International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Enhancing Demand Forecasting Performance Using Deep Learning and Time Series Data Augmentation Techniques

Jinseop Yun[1], Yejun Park[2], Doohee Chung*[3]

Submitted: 14/08/2025 Revised: 25/09/2025 Accepted: 05/10/2025

Abstract: Accurately forecasting demand remains a persistent challenge for organizations, especially under conditions of high uncertainty and data scarcity. While machine learning and deep learning methods have advanced beyond traditional statistical approaches, their effectiveness is often constrained by limited data availability. To address this critical issue, this paper introduces an innovative and systematic framework that integrates advanced time series data augmentation techniques, the Long Short-Term Memory (LSTM) deep learning model, and Average Demand Interval-Coefficient of Variance (ADI-CV) methodology. The proposed framework leverages ADI-CV to categorize time series patterns, enabling the application of tailored augmentation techniques such as Moving Block Bootstrap (MBB), Time-Conditional GAN (T-CGAN), and Transformer-based Time-Series Conditional GAN (TTS-CGAN). These techniques ensure the generation of synthetic data that accurately reflects temporal characteristics and market conditions, overcoming the traditional limitations of data scarcity. Our experimental results demonstrate that the augmented time series data significantly enhances forecasting performance across diverse and complex demand scenarios. This framework not only addresses the critical gap in demand forecasting methodologies but also establishes a scalable and adaptable solution for enterprises operating in volatile and dynamic market environments. By offering a robust tool to improve predictive accuracy and reliability, this study contributes a novel methodology with the potential to transform business decision-making processes.

Keywords: *Time Series Data Augmentation; Deep Learning; Demand Forecasting, ADI-CV*

1. Introduction

Accurate prediction of product sales and shipments has been a significant challenge in corporate management. Precise demand forecasting forms the basis for effective inventory management, production planning, and promotional activities [1], enhancing a company's productivity and profitability [2]. Scholars emphasize that improving the accuracy of demand forecasting plays a critical role in enhancing corporate competitiveness, especially amid increasing market complexity and volatility. Accurate forecasts enable companies to optimize production planning, reduce inventory costs, and swiftly respond to fluctuating customer demands in unpredictable markets [1,3,4].

Demand forecasting research has long relied on traditional statistical methods, including regression models such as Linear Regression, Lasso Regression, and Ridge Regression, as well as time series models like ARIMA and SARIMA [5]. While these methodologies are somewhat effective in predicting demand patterns using historical data, they often require extensive manual parameter tuning and significant domain knowledge, making them time-consuming and less adaptable to rapidly changing market conditions.

To overcome these limitations, deep learning models have recently been applied to demand forecasting. Deep Neural Networks (DNNs), such as Multi-Layer Perceptrons (MLPs), have been used to model complex patterns [6] but may struggle with sequential data. Recurrent Neural Networks (RNNs) [7] address this by capturing temporal dependencies, yet they can suffer from issues like vanishing gradients. The Long Short-Term Memory (LSTM) model, an advanced type of RNN, effectively mitigates these issues and has shown excellent performance in solving complex demand forecasting problems across various fields [8,9,10].

Recent research trends focus on developing hybrid models that integrate various techniques to overcome the limitations of single machine learning or deep learning models [11,12]. Representative examples include combinations of LSTM with traditional time series forecasting models, CNN-LSTM hybrid models, ensembles of deep learning and machine learning techniques, and attention-based models [13,14,15]. These models are gaining attention because they can better capture spatiotemporal characteristics and long-term and short-term data dependencies that existing models could not.

However, these advancements have not addressed a fundamental challenge in demand forecasting: data scarcity [16]. While hybrid models have enhanced modeling capabilities, they remain heavily dependent on the quality and quantity of input data, which limits their effectiveness in scenarios with insufficient or inconsistent

1 Impactive AI, Seoul – 04516, Republic of Korea

ORCID ID : 0009-0006-2240-9327

2 School of Applied Artificial intelligence, Handong Global University, Pohang – 37554, Republic of Korea

ORCID ID : 0009-0006-7599-2069

3 School of Applied Artificial intelligence, Handong Global University, and Impactive AI, Pohang – 37554, Republic of Korea

ORCID ID : 0000-0002-7413-0778

** Corresponding Author Email: profchung@handong.edu*

data [17]. This limitation underscores the need for innovative solutions that focus not only on model sophistication but also on improving the availability and diversity of training data.

According to [18], 52% of companies worldwide face significant challenges in adopting AI solutions due to insufficient data. Since machine learning is inherently data-intensive, data scarcity imposes severe constraints on the application of machine learning-based demand forecasting models. Insufficient data reduces training opportunities for ML models, leading to diminished performance and significantly impairing their ability to generalize effectively in forecasting scenarios.

In this context, data augmentation techniques play a crucial role in addressing these challenges. By supplementing insufficient data and generating enriched and diverse datasets, data augmentation mitigates the limitations of data scarcity and significantly enhances forecasting performance. Resolving data scarcity opens up the potential to apply machine learning forecasting models to the remaining 50% of demand forecasting scenarios that were previously inaccessible due to data limitations.

To address these challenges in demand forecasting, this study highlights data augmentation as a hybrid approach that offers a direct and effective solution. Unlike traditional methods of information expansion, such as external data integration or transfer learning, data augmentation directly tackles dataset limitations by generating synthetic variations [19]. By enhancing the quantitative and qualitative diversity of existing data, it enables models to encounter a broader range of potential data patterns beyond what is typically observed in real-world settings, thereby improving model robustness and adaptability [20]. In the context of product demand forecasting—which must adapt to challenges such as new product launches, seasonal variability, and shifting market trends— diversity learning through data augmentation is essential [21].

This study distinguishes itself by presenting a novel approach that integrates data augmentation techniques with deep learning models to address the dual challenges of data scarcity and prediction accuracy in demand forecasting scenarios. Unlike prior research, which has predominantly focused on enhancing model complexity or optimizing parameters, this study directly tackles the issue of data scarcity by generating synthetic data, enabling robust learning even in environments with limited data availability.

Data augmentation, however, involves more than merely increasing the quantity of data; it requires a sophisticated design that accounts for the unique characteristics of each dataset. Randomly applied augmentation techniques can destabilize the training process or degrade model performance. To mitigate these risks, this study employs the Average Demand Interval-Coefficient of Variance (ADI-CV) methodology, systematically analyzing the variability and frequency of data to accurately select augmentation techniques optimized for each demand pattern. By doing so, the study maximizes the effectiveness of data augmentation, improving both the predictive accuracy and robustness of the models.

Therefore, this study introduces augmentation models tailored to diverse demand patterns, optimizing the performance of LSTM-based forecasting models. This approach not only addresses the often-overlooked issue of data scarcity in previous research but also presents a modeling framework with scalability and adaptability to dynamic and volatile market environments. Consequently, this study significantly enhances the accuracy and reliability of demand forecasting while providing a robust foundation for improving the quality of decision-making processes.

2. Literature Review

2.1. Approaches to Demand Forecasting

In demand forecasting research, there is a notable emphasis on the capability of deep learning models to produce highly accurate predictions, even when working with univariate time series data [22]. Unlike traditional statistical models, deep learning techniques effectively capture complex nonlinear patterns and temporal dependencies inherent in univariate data. Traditional models often assume linearity and may struggle with the non-stationarity and noise present in real-world demand data. In contrast, deep learning models like LSTM can learn long-term dependencies and nonlinear relationships without extensive manual parameter tuning [23,24]. This advantage is particularly prominent in situations with high data complexity and volatility, where capturing intricate patterns in univariate time series is crucial for accurate forecasting.

Deep learning-based time series forecasting models began with the MLP. However, MLPs lack recurrent structures, making them ineffective at capturing temporal dependencies in time series data [25]. To address this issue, RNNs were introduced, which learn patterns in sequential data through recurrent connections [7]. Despite this advancement, RNNs face challenges in learning long-term dependencies due to the vanishing and exploding gradient problems during training [26]. To overcome these limitations, the Gated Recurrent Unit (GRU) was developed, enhancing model efficiency by regulating the flow of information with gating mechanisms [27]. [28] demonstrated that GRUs effectively process sequences of various lengths by sharing the same parameters across time steps.

Finally, the LSTM network introduced additional gates and cell states, allowing for more stable learning of long-term dependencies [29]. [30] revealed that LSTM outperforms traditional models such as ETS, ARIMA, SVM, and standard RNNs in forecasting complex univariate time series data. This progression from MLP to RNN, GRU, and LSTM has significantly enhanced the ability to model complex and irregular time series data [31].

Despite significant advancements in the application of deep learning in demand forecasting, several critical limitations remain in existing research. First, time series data used in demand forecasting often exhibits diverse and complex patterns such as seasonality, trends, and cyclicality. Accurately modeling these patterns requires advanced statistical and domain knowledge as well as sophisticated modeling techniques. Although deep learning models capture complex nonlinear relationships by utilizing layered architecture and nonlinear activation functions such as ReLU, sigmoid, and tanh that enable them to model intricate patterns in data, relying on a single predictive model may not always yield satisfactory results when tackling intricate sales forecasting problems due to the high complexity and variability inherent in demand data [32].

To address this challenge, recent studies have increasingly reported advanced hybrid models designed to overcome the limitations of single deep learning models. These hybrid models optimize prediction performance by integrating multiple processes such as data preprocessing, parameter tuning, clustering, error correction, and postprocessing into a cohesive framework. Such models consistently outperform single models, as demonstrated by various research efforts. For example, [33] introduced a hybrid model combining Seasonal and Trend Decomposition using Loess (STL) with a Duo-Attention Deep Learning Model (DADLM) for tourism demand forecasting. This STL-DADLM model mitigated overfitting and achieved higher accuracy compared to traditional

models, showcasing the advantage of integrating decomposition and deep learning techniques. Similarly, [34] proposed a (c, l)-LSTM+CNN model for power demand forecasting, where the (c, l)-LSTM extracts temporal features, and the CNN generates refined prediction profiles. This hybrid approach outperformed ARIMA and other models, particularly excelling in short-term forecasts with fine temporal granularity.

However, the second and third challenges remain unresolved by hybrid models.

Second, deep learning models are at risk of overfitting the training data, which can hinder the model's ability to generalize to new data [35]. Overfitting is a major factor that can significantly reduce the predictive performance of a model in real-world operational environments. Hybrid models, while enhancing predictive capabilities, may still suffer from overfitting, especially when they become overly complex or when the amount of training data is limited.

Third, deep learning models require a sufficient amount of data to be effectively trained. The study by [17] demonstrated that data scarcity is a major factor that diminishes the performance of LSTM models, arguing that with sufficient data, these models could outperform traditional forecasting methods. However, in real-world scenarios, obtaining enough data is often challenging, especially in the case of new products where data may be nonexistent [36]. This issue critically limits the application scope of deep learning models, making it difficult to generate accurate forecasts for unique market conditions or niche products [37]. Hybrid models do not fundamentally solve the problem of data scarcity, as they often require even more data to train the additional components effectively.

Therefore, while hybrid models can address the first challenge by enhancing the ability to model complex demand patterns, they do not effectively resolve the issues of overfitting and data scarcity. These limitations necessitate alternative approaches that can mitigate overfitting and improve model performance even when data is limited.

2.2. Data Scarcity and Time Series Data Augmentation

Data scarcity remains a significant challenge in the development of deep learning models. While model-centric techniques like dropout, batch normalization, and transfer learning help mitigate overfitting and enhance generalization, they do not fundamentally resolve limitations in data quantity and quality. The most effective way to overcome these constraints is to increase data availability.

In this context, [19] emphasize that data augmentation is a foundational and effective approach to overcome overfitting caused by limited data. Data augmentation artificially enhances dataset diversity by applying transformations to existing data, creating new samples without additional data collection. Initially starting with simple transformation techniques in the field of image recognition, it has evolved into advanced, deep learning-based methods such as style transfer and Generative Adversarial Networks (GANs) [38,39]. This approach has established itself as a powerful solution in various fields, addressing data scarcity and improving model performance.

In the field of time series forecasting, there has been active research on data augmentation to address challenges such as data scarcity, imbalanced data distribution, and privacy concerns [40,41,42]. [20] categorized time series data augmentation into basic approaches (cropping, flipping, jittering) and advanced techniques (decomposition, statistical generative models, machine learning-based methods), each aiming to enhance prediction accuracy.

A variety of methods have been proposed to augment time series data effectively. Time Series Bootstrapping increases data diversity by resampling the original data, thus helping to estimate prediction uncertainties [43]. Dynamic Time Warping Barycentric Averaging (DBA) combines multiple time series to create a representative series that accounts for temporal variations [44]. Markov Chain Monte Carlo (MCMC) simulates realistic time series by estimating parameters of probabilistic models [45], while GRATIS generates synthetic data with diverse characteristics, improving model performance [46]. Additionally, Variational Auto-Encoders (VAE) and Generative Adversarial Networks (GANs) leverage deep learning to generate data that closely resembles real time series, significantly expanding datasets without manual collection [47, 48].

[49] demonstrated that RNN-based models trained on data augmented with methods like MBB, DBA, and GRATIS outperformed traditional univariate models. This suggests that data augmentation can significantly enhance both the accuracy and generalization capability of time series forecasting models in data-scarce environments.

However, data augmentation does not always guarantee improved predictive performance. The effectiveness of augmented data depends on how well it captures the complexity and diversity of the original dataset. If augmentation introduces distributional shifts or adds unnecessary noise, it may hinder the model from learning essential patterns and relationships [42]. For example, [50] tested 12 time series augmentation methods in classification tasks but found that models like LSTM-FCN and MLP did not always benefit from augmented data. This suggests that augmentation needs to align closely with both data characteristics and model architecture to be effective.

In demand forecasting, successful augmentation requires techniques tailored to the specific demand patterns present in the data. Understanding and decomposing data types before applying augmentation can improve the relevance of augmented data. [51] highlighted that traditional GAN-based augmentation performs poorly with non-uniform data distributions. To address this, they proposed a Decomposition-based Data Augmentation Scheme (DAST), which separates data into daily-load, seasonal context, and irregular components for targeted augmentation. This approach significantly reduced forecasting errors in building load predictions compared to conventional methods, illustrating the importance of aligning augmentation strategies with data characteristics.

In demand forecasting, domain-specific metrics like ADI and CV are often used for data classification [52]. These metrics have also been leveraged to apply tailored predictive models, as shown by [53]. For example, [54] improved forecasting accuracy by segmenting data according to demand patterns and applying appropriate predictive models to each segment using a Dynamic Weighting Strategy (DWS). However, research on applying augmentation techniques based on demand patterns remains limited, suggesting a need for further exploration.

2.3. Ensembled approach for forecasting

This study employs ADI-CV analysis to classify demand patterns for effective data augmentation. ADI measures the average interval between demand occurrences, indicating the intermittent demand, while CV reflects the variability in demand volume. By combining these indicators, ADI-CV analysis enables segmentation based on demand frequency and variability, providing a structured approach to handling diverse demand patterns [52].

Previous research has highlighted the practical applications of

ADI-CV analysis. For instance, [55] developed an optimization strategy for managing spare parts with irregular demand patterns using ADI-CV, and [56] optimized inventory policies by analyzing demand frequency and variability to minimize costs. These studies demonstrate the effectiveness of ADI-CV analysis in operational decision-making, offering insights that can enhance efficiency in corporate inventory and demand management.

In this study, ADI-CV analysis is utilized not only to classify demand patterns but also as a foundation for selecting appropriate data augmentation techniques. By aligning augmentation methods with specific demand characteristics, this approach aims to improve demand forecasting accuracy and model robustness across varied demand scenarios.

In conclusion, this literature review confirms that properly analyzing demand patterns and applying suitable data augmentation techniques can enhance the performance of deep learning models. Therefore, this paper applies the ADI-CV methodology as a systematic tool for demand pattern analysis and explores various augmentation techniques such as Moving Block Bootstrap (MBB), Time-Conditional GAN (T-CGAN), and Transformer Time-Series Conditional GAN (TTS-CGAN) to address data scarcity issues. Through this, the study aims to develop a hybrid model that combines these augmentation techniques with the LSTM prediction model to improve performance.

3. Methodology

3.1. Data Augmentation Model

Generative Adversarial Networks (GANs), introduced by [38], have emerged as a powerful framework for generating synthetic data by learning the underlying distribution of real datasets. GANs consist of two neural networks, the generator and the discriminator that are trained simultaneously through an adversarial process. The generator creates synthetic data samples, while the discriminator evaluates their authenticity, guiding the generator to produce data that closely resembles the real data.

In the context of time series data, traditional GANs face challenges due to the sequential and temporal dependencies inherent in such data. To address these issues, specialized GAN architectures have been developed to handle time series data more effectively.

3.1.1. T-CGAN

T-CGAN is proposed as a methodology to generate new data in cases where time series data is irregularly collected or insufficient [57]. This model uses GAN architecture consisting of two components: a Generator (G) and a Discriminator (D). Specifically, it employs a Conditional Generative Adversarial Network (CGAN) architecture, where the generator and discriminator learn to generate and distinguish data based on given conditions.

The Generator (G) creates new data based on the provided conditions, while the Discriminator (D) distinguishes whether the data is real or generated. Through this process, the generator progressively produces data that closely resembles real data to deceive the discriminator. The generator is implemented using a deconvolutional neural network, and the discriminator is implemented using CNN. By conditioning on the time information that indicates the data collection points, T-CGAN generates time series data that mimics the actual data distribution.

The model is primarily composed of three spaces: noise vector space (Z), time information space (T), and data space (X). The noise vector space (Z) includes noise vectors used as input values for the model, sampled from a Gaussian distribution to generate new time series data. The time information space (T) contains information indicating the collection time of each data point, which is provided as a condition to both the generator and the discriminator. The data space (X) includes both the actual time series data and the generated time series data. Through the interaction of these spaces, the model generates data that mimics the actual data distribution $p_{data}(x,t)$ based on the given time information condition.

The objective function is as follows:

$$\min_G \max_D V(D,G) = E_{x\sim p_{\text{data}}(x)}[\log D(x \mid t)] + E_{z\sim p_z(z)}\left[\log\left(1 - D(G(z \mid t))\right)\right] \quad (1)$$

The model operates with the goal of enabling the discriminator model D to accurately distinguish between real and generated data, while the generator model strives to deceive the discriminator by producing data indistinguishable from the real data. Here, $t = \langle t_1, \ldots, t_n \rangle$ is an ordered vector of timestamps randomly sampled from T. The model can also generate new time series corresponding to timestamps that do not present in the training set.

The generator network takes noise vectors and timestamps as inputs to generate time series data similar to real data. This is accomplished using four transposed convolutional layers. Each layer applies ReLU activation functions and batch normalization, except for the last layer, to maximize learning efficiency and stability of the network. The generator network is defined as follows:

$$G: (Z,T) \rightarrow X \quad (2)$$

The discriminator network receives real data and generated data along with their corresponding timestamps as inputs to determine whether the data is real or generated. This network is composed of two convolutional layers, max-pooling layers, and a final fully connected layer that makes the final discrimination decision. The discriminator network is defined by the following function:

$$D: (X,T) \rightarrow [0,1] \quad (3)$$

3.1.2. TTS-CGAN

TTS-CGAN [58] is also an effective augmentation technique designed to address the problem of insufficient time series data. Traditional GAN models were limited to generating data for a single label, but TTS-CGAN introduces a label embedding strategy to generate conditional time series data that incorporates label information. By incorporating a transformer-based GAN structure, TTS-CGAN captures complex temporal patterns such as long-term dependencies, enabling the generation of highly accurate data.

This model also consists of two main components: the generator and the discriminator. The generator starts with a random vector from the latent space and uses a multi-head self-attention mechanism and GELU activation function within an MLP to transform it into a sequence that matches the length of the time series data but with many hidden dimensions. The discriminator distinguishes between generated data and real data, leveraging a transformer encoder to learn the temporal order and patterns of the data.

TTS-CGAN operates by integrating conditional information into the traditional GAN framework. It combines Adversarial Loss and Categorical Loss to train the model so that the generated data is

indistinguishable from real data while accurately reflecting the target category labels. Additionally, to assess the similarity between generated and real data, it introduces the Wavelet Coherence Similarity metric, which maximizes the variance between classes and minimizes the variance within classes. This metric quantitatively evaluates the similarity between actual signals and generated signals, ensuring that the generated data accurately reflects the characteristics of the original data.

$$L_{adv} = E[\log D_{adv}(x)] + E\left[\log\left(1 - D_{adv}(G(z,c))\right)\right] \quad (4)$$

$$L_{cls} = E[-\log D_{cls}(c \mid x)] \quad (5)$$

$$ucoh = \frac{\left|S\left(C_x(a,b) \cdot C_y(a,b)^*\right)\right|}{\sqrt{S(|C_x(a,b)|^2) \cdot S\left(\left|C_y(a,b)\right|^2\right)}} \quad (6)$$

3.1.3. MBB

Bootstrap is a critical method in statistical inference for estimating the distribution of statistics without relying on parametric assumptions [59]. MBB is a bootstrap method tailored for time series data, designed to preserve the time dependence structure of the original time series by maintaining the order of data within the same block [60]. Bootstrap-based data augmentation techniques that generate time series similar to the original dataset distribution have proven effective in improving model accuracy [49].

The core of MBB is to generate new samples that reflect the autocorrelation within the time series data. According to [61], the process of selecting blocks of consecutive observations and the length of these blocks directly affect the accuracy of bootstrap estimates. Therefore, the appropriate block length should be determined by balancing the autocorrelation characteristics of the data and the accuracy of the estimates obtained through resampling. The mechanism of MBB is as follows [43]:

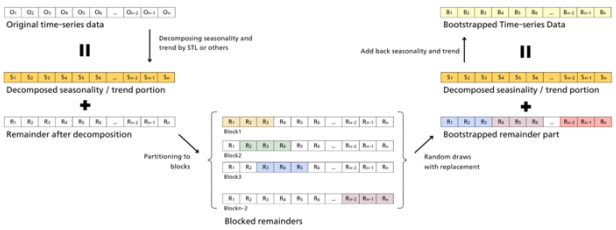

Fig. 1. MBB Architecture

Decomposition: Decomposing data D_t into trend T_t, seasonality S_t , and residual R_t components is essential for clearly understanding the fundamental structure of the data. Decomposition using STL flexibly captures the nonlinear trends and seasonal variations within the data. This decomposition process reduces the complexity of the data for future analysis and provides a foundation for evaluating the influence of each component separately.

$$D_t = T_t + S_t + R_t \quad (7)$$

Block Generation: Generating blocks of consecutive observations from the residual R_t is the central step of MBB. The block length l should adequately reflect the autocorrelation properties of the data. According to Bergmeir's suggestion, the block length can be adjusted based on the data type and the analysis purpose [43]. This process plays a crucial role in ensuring the diversity and reliability of the bootstrap samples.

Sample Extraction and Recombination: Randomly extracting the reconstructed residual blocks and adding them to the existing trend and seasonality generates new synthetic time series data. This process is crucial for creating new data samples while maintaining the statistical characteristics of the original data. By applying the resampling principle that preserves the temporal structure of the data, this step contributes to enhancing the reliability of the analysis.

$$D_t{}^* = T_t + S_t + R_t{}^* \quad (8)$$

Estimation Calculation: Estimating the statistics or model parameters of interest from the reconstructed data $D_t{}^*$ involves utilizing the samples obtained through the MBB approach to derive the statistical estimate distribution of the analysis target. Through this iterative process, confidence intervals, standard errors, and other evaluation metrics of the estimates can be assessed. This step verifies the reliability and predictive power of the model, effectively managing the uncertainty and variability of the data.

3.2. Demand Pattern Analysis: ADI/CV

In this study, ADI and CV are applied to distinguish demand patterns. ADI measures the average interval between demand occurrences for an item or service within a specific period, indicating demand frequency. CV assesses the variability in demand, reflecting its stability and predictability. These two indicators quantify the diversity in demand intervals and sizes, categorizing product demand patterns into four types: Smooth, Intermittent, Lumpy, and Erratic [62].

ADI represents the average time interval between consecutive demands. By calculating ADI, the frequency of demand occurrence can be understood, providing crucial information for inventory management and demand forecasting. A high ADI indicates long intervals between demand occurrences, which may suggest an irregular demand pattern. Here, the time interval between the i-th and $i+1$-th demands is t_I, and N is the total number of demand occurrences within the measurement period.

$$ADI = \frac{\sum_{i=1}^{N} t_i}{N} \quad (9)$$

CV is the value obtained by dividing the standard deviation σ by the mean demand μ, representing the relative variability of demand. A high CV indicates a wide variation in demand sizes, which can complicate demand forecasting and inventory management for the product. Here, D_I is the size of the i-th demand, and D is the average demand size.

$$CV = \frac{\sqrt{\frac{\sum_{i=1}^{N}(D_i - D)^2}{N}}}{D}, \; where\; D = \frac{\sum_{i=1}^{N} D_i}{N} \quad (10)$$

Combining ADI and CV allows for a precise understanding of demand patterns for products. For instance, when both ADI and CV values are high, it indicates that demand is highly irregular and difficult to predict. The study by [52] established threshold values of CV 0.49 and ADI 1.32 to categorize demand patterns. Based on these thresholds, the four classified demand patterns are as follows: When ADI is less than 1.32 and CV is also less than 0.49, a Smooth demand pattern is observed, indicating a regular and predictable demand pattern. Conversely, when ADI increases to 1.32 or more while CV remains below 0.49, an Intermittent demand pattern is observed, characterized by infrequent but stable demand

quantities. When ADI is less than 1.32 and CV exceeds 0.49, a Lumpy demand pattern emerges, where the occurrence frequency is irregular, and the demand size is highly variable. Lastly, when both ADI and CV exhibit high values, an Erratic demand pattern is observed, making it difficult to predict and requiring considerable flexibility and caution in inventory management.

Table 1. Classification of Demand pattern in terms of ADI-CV

ADI	CV	Demand Pattern
0.0 < ADI < 1.32	0.0 < CV < 0.49	Smooth
1.32 < = ADI	0.0 < CV < 0.49	Intermittent
0.0 < ADI < 1.32	0.49 < = CV	Lumpy
1.32 < = ADI	0.49 < = CV	Erratic

3.3. LSTM

To address the complexities inherent in demand forecasting, this study employs the LSTM model, a specialized Recurrent Neural Network (RNN) architecture proposed by [29]. The LSTM model is chosen due to its ability to capture long-term dependencies within time series data, a characteristic essential for demand forecasting. Unlike traditional RNNs, LSTM mitigates the gradient vanishing issue, enabling effective learning over extended temporal sequences [63,64].

The main components of the LSTM model are the hidden state and cell state. The hidden state processes short-term information, while the cell state stores long-term information. These states manage the flow of short-term and long-term information, constituting the core mechanism that determines the model's performance. LSTM operates through three main gates: input gate, forget gate, and output gate.

Input Gate decides whether new information should be added to the memory cell by considering the current input and previous hidden state. Forget Gate removes unnecessary information using the previous cell state and current input. Output Gate determines the output information based on the current cell state, adjusting it to generate the final output.

This structure of LSTM effectively reflects the characteristics of time series data, which occurs sequentially over time and has strong dependencies between time points. Time series data includes both short-term fluctuations and long-term trends, requiring a model capable of handling these aspects.

LSTM stores long-term information through the cell state, updating or maintaining it as needed. This mechanism is highly effective for learning long-term patterns in time series data. Additionally, the input gate and forget gate help remove unnecessary short-term fluctuations and selectively remember important information, aiding the model in learning key data characteristics.

By managing information flow through its gates, the LSTM model effectively captures the dynamic and sequential nature of demand forecasting data. This architecture enables it to learn complex temporal patterns, making it highly suitable for predicting future demand based on historical trends. Consequently, the LSTM model offers a robust framework for handling time-dependent patterns, supporting accurate long-term demand predictions.

In summary, LSTM's ability to retain crucial historical information over extended periods while filtering out irrelevant data makes it an optimal model for time series forecasting. These attributes enhance its predictive accuracy and reliability, establishing it as a powerful tool for learning long-term dependencies in demand forecasting applications.

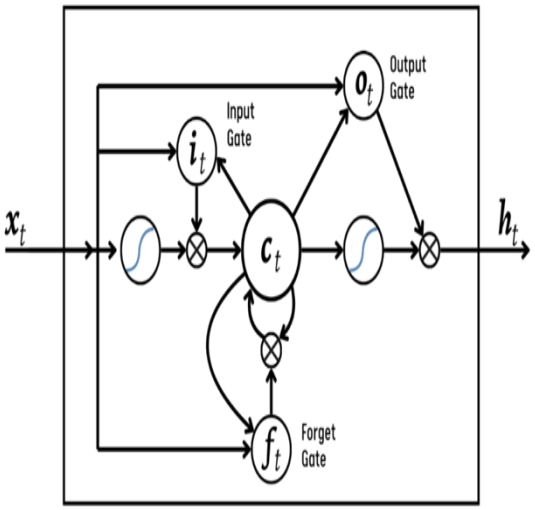

Fig. 2. LSTM Architecture

4. Demand Forecasting Model Combined with Data Augmentation Techniques

This study proposes a hybrid model that combines data augmentation techniques and deep learning models to achieve accurate product demand forecasting in situations with limited data. The framework explaining the procedure for this model is shown in Figure 3.

4.1. Identifying Demand Pattern through ADI/CV Analysis

We calculated the ADI and CV to analyze the frequency and variability of demand for each product. Based on these metrics, products were classified into one of four demand patterns: 'Smooth,' 'Erratic", "Intermittent,' or 'Lumpy.'

4.2. Data Augmentation through various Models

Subsequently, data with similar patterns to the actual ones is generated for each demand pattern using various augmentation models. The generated data undergoes a normalization process and is pooled together with the existing data for model training.

4.3. Training LSTM for each Demand Pattern

We trained the time series prediction model using LSTM on the pooled data, which includes both the augmented and original data generated for each demand pattern. Hyperparameter tuning is the process of adjusting the model's hyperparameter values to optimize its performance. For a base learning model like LSTM, it is essential to set various hyperparameters appropriately, such as LSTM cell dimensions, the number of epochs, hidden layers, mini-batch size, and normalization layers. Therefore, we used Keras-Tuner to test multiple hyperparameter combinations and automatically determine the optimal values.

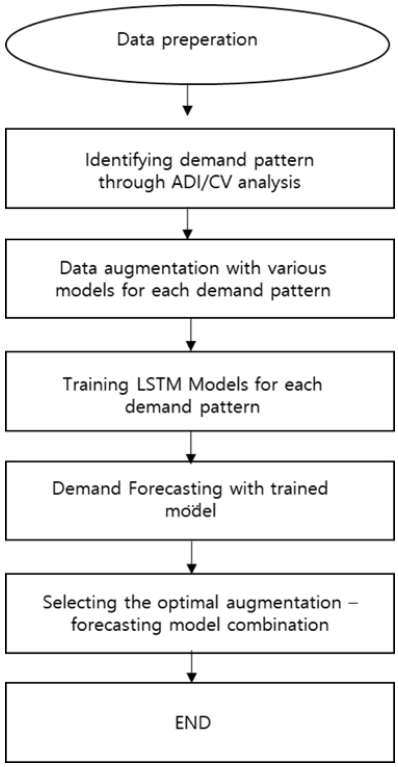

Fig. 3. Experiment framework

4.4. Demand Forecasting with Trained Model

In the demand forecasting stage, use the trained LSTM model to predict the future demand for each item. Evaluate the model's performance across the entire dataset and for each demand pattern and identify the most suitable prediction model for each pattern.

4.5. Selecting Optimal Augmentation-Forecast Model Combination

In the optimal combination selection process, compare and analyze the predictive performance of various data augmentation techniques and prediction model combinations. Determine the most effective data augmentation-prediction model combination for each demand pattern. Use performance evaluation metrics such as nRMSE (Normalized Root Mean Square Error), nMAE (Normalized Mean Absolute Error), and sMAPE (Symmetric Mean Absolute Percentage Error) to analyze the performance of each combination. Select the model combination with the highest performance for each demand pattern as the optimal model.

5. Data

The data analyzed in this study consists of weekly demand data spanning a total of 243 weeks, from the first week of 2019 to the fourth week of August 2023, from a global pharmaceutical company. Out of 369 products, 168 items were identified as the target for prediction. Initially, 248 items with confirmed demand variability were selected through ADI/CV analysis. Subsequently, the distribution of shipment frequencies was examined to select the final items for the model. The distribution based on shipment frequencies showed that 40 items (approximately 16.12%) were fewer than 5 shipments, and 55 items (approximately 24.38%) were fewer than 10 shipments. The items used for model training were those with 40 or more shipments, totaling 168 items (approximately 67.74%). The average delivery quantity for these items was 4288.92, with a minimum value of 0.0 and a maximum value of 704,100.0, showing a wide range.

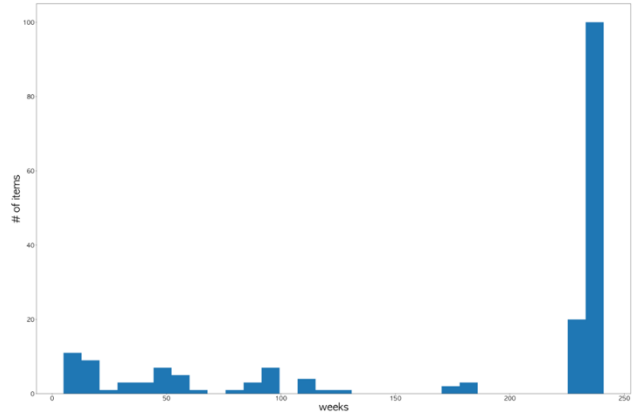

Fig. 4. Weekly data distribution

A descriptive statistical analysis was conducted on the weekly demand data. The analysis was based on the entire dataset from the first week of 2019 to the fourth week of August 2023, encompassing a total of 179,292 records.

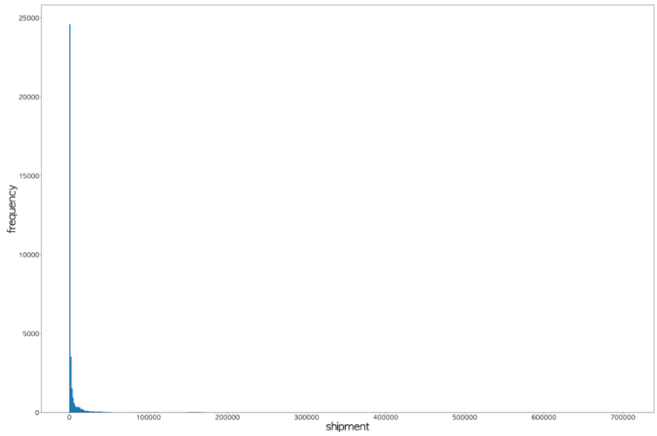

Fig. 5. Data distribution based on shipment frequencies

The first quartile (Q1) of the demand data is 0, indicating that at least 25% of the data points are 0. The median (50th percentile) is 72, while the mean demand is approximately 4,289, significantly higher than the median. This suggests that extreme demand values are substantially elevating the mean. The third quartile (Q3) is 1,000, indicating that most data points have lower demand than the mean. The interquartile range (IQR) is 1,000, which is smaller than the mean, indicating that the data is highly clustered around the median with low variability. The maximum demand value is 704,100, showing a highly skewed distribution.

CV is approximately 5.11, indicating that the standard deviation is much larger than the mean, signifying high variability in the data. The skewness is about 11.89, showing that the demand distribution is heavily skewed to the right, with most demand being low but some extremely high. The kurtosis is around 196.43, indicating that the demand distribution is much more peak than a normal distribution, with many extreme values.

Table 2. Descriptive statistical analysis on weekly demand data

Number of Data Points	**179,292**
1st Quartile	0
Median	72
Mean	4289
3rd Quartile	1000

Interquartile Range	1000
Maximum	704100
Coefficient of Variation	5.11
Skewness	11.89
Kurtosis	196.43

This statistical analysis provides crucial insights into the fundamental characteristics of the data, which is important for the modeling process. High skewness and kurtosis indicate the possibility of extreme demand occurrences, and the high CV implies significant demand variability. Therefore, it is essential to design a model that can appropriately understand, reflect, and predict these data characteristics and distributions.

6. Experimental Results

6.1. Demand Pattern

In this section, ADI and CV were utilized to analyze the demand patterns of products. First, data was collected that measured the demand quantity D_I during the period t_I for each item. By calculating the average of these time intervals, the ADI was derived. The axis in Figure 6 represents time, and the demand occurring at specific points in time is indicated as D. Additionally, to evaluate the variability in demand size, the standard deviation of the demand quantities D_i (i=1,2,3,4,…,n) was calculated, and this was divided by the mean demand to obtain CV.

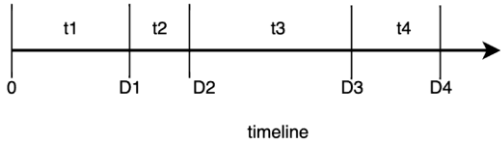

Fig. 6. Example of demand occurrences between each interval

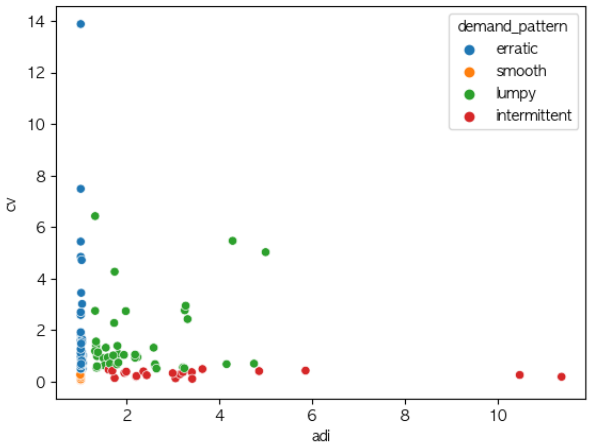

Fig.7. Demand pattern analysis based on ADI-CV

Based on the demand occurrence times and quantities for 168 items, ADI and CV were calculated and plotted in Figure 7, with ADI on the x-axis and CV on the y-axis. Using specific threshold values, the items were categorized into four types: Smooth, Erratic, Intermittent, and Lumpy. The classification resulted in 60 items (35.71%) as Smooth, 60 items (35.71%) as Erratic, 16 items (9.52%) as Intermittent, and 32 items (19.04%) as Lumpy.

Smooth items exhibit stable demand with a low CV of 0.12, slight right skewness (0.77), and kurtosis of 1.82. They have an average demand of 243 units and minimal zero-demand occurrences (1.22%). Erratic items show high variability with a CV of 0.86, high skewness (3.33), and kurtosis of 16.40, indicating extreme fluctuations, with an average demand of 241 units and a low zero-demand frequency of 2.45%. Intermittent items display irregular demand, characterized by a CV of 0.34, right skewness (0.92), and low kurtosis (0.03), with frequent zero demand (61.41%) and a lower average demand of 113.9 units. Lumpy items exhibit sporadic demand with a CV of 1.03, high skewness (2.07), and kurtosis of 4.65, along with a substantial zero-demand frequency of 44.95% and an average demand of 129.4 units.

This analysis provides critical insights for developing tailored forecasting models that address the distinct characteristics of each demand pattern.

6.2. Data Augmentation

In this section, we explain the process of identifying individual product demand patterns through ADI-CV analysis and forming clusters based on these patterns. For each cluster, we use t-SNE to map the data into a 2D space, visually verifying the effectiveness of data augmentation. Additionally, we quantitatively evaluate the distribution differences between augmented data and original data using the KL divergence metric and analyze the consistency of model results through epistemic indicators. This allows us to numerically confirm how accurately the augmented data mimics the original data distribution. This section details the visual and quantitative validation methods for data augmentation and comprehensively analyzes the impact of data augmentation on improving the accuracy and reliability of the demand forecasting model.

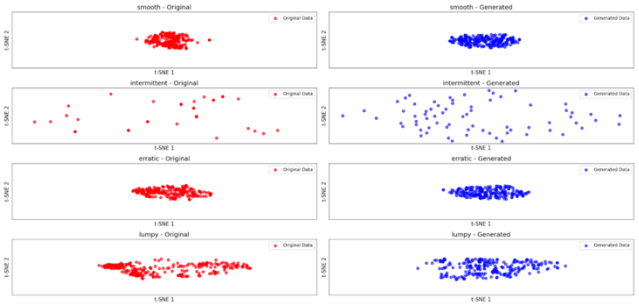

Fig. 8. Comparison of distribution

We visually validated the effectiveness of data augmentation by comparing the distributions of the original data (in red) and the data generated through augmentation (in blue). The original data forms distinct clusters according to the Smooth, Erratic, Intermittent, and Lumpy patterns, and the augmented data closely mimics these patterns. The distribution of the augmented data appears to be very similar to that of the original data.

In this study, we used Kullback-Leibler Divergence (KL divergence) to quantitatively assess how closely the augmented data approximates the original data distribution. KL divergence is an asymmetric measure that quantifies the difference between two probability distributions, helping to determine how well the new data generated by the augmentation technique retains the original data distribution. The KL divergence between two probability distributions P and Q is defined as follows:

$$KL(P \parallel Q) = \sum_{x} P(x) \log\left(\frac{P(x)}{Q(x)}\right) \tag{11}$$

In this equation, $P(x)$ represents the probability distribution of the

original data, while $Q(x)$ represents the probability distribution of the augmented data. A lower KL divergence value indicates that the distribution $Q(x)$ of the augmented data closely approximates the distribution $P(x)$ of the original data.

Along with KL divergence, Epistemic Indicators provide another measure of the reliability of the augmented data. Epistemic Indicators calculate the conditional probability of each augmented data point given the prediction results. By averaging these individual conditional probability values, the overall average reliability of the augmented dataset is determined. A higher average value indicates that the augmented data maintains the characteristics of the original data while incorporating sufficiently diverse information. This average conditional probability is used to assess the reliability of the augmented dataset. A high reliability score suggests better generalization and predictive performance when the augmented data is used for model training. The formula for calculating the Epistemic Indicators is as follows:

$$\hat{\mu}_i = \frac{1}{E}\sum_{e=1}^{E} p_{\theta(e)}(y_i^* \mid x_i) \tag{12}$$

In this context, $\hat{\mu}_i$ represents the conditional mean prediction value for the i-th data point. This value is calculated as the average of the prediction values obtained from multiple augmented data points. $p_{\theta(e)}(y_i^* \mid x_i)$ denotes the conditional probability of the predicted value y_i^* given the input x_i using the model θ for the e-th augmented data point. This probability indicates how likely the predicted result is for that specific data point. E represents the number of augmented data points, and it is used to average the probabilities across all augmented data.

Table 3. Model results on KL divergence and Epistemic

Model	KL Mean	KL STDV	Epistemic
MBB	0.18	0.052	95.14
TC-GAN	0.9	0.162	95.56
TTS-CGAN	1.45	0.586	95.85

The results of the KL divergence play a crucial role in understanding the impact of data augmentation on model performance. Specifically, the MBB model shows a low average KL value of 0.08, indicating that the augmented data closely reflects the distribution of the original data. In contrast, the TTS-CGAN model has a high average KL value of 1.42, suggesting that the augmented data from this model exhibits a greater discrepancy from the original data.

6.3. Results

This study aims to compare the performance of four major models in the field of demand forecasting. The target models are Moving Block Bootstrap LSTM (MBB-LSTM), Time-Conditional Generative Adversarial Network LSTM (TCGAN-LSTM), Transformer Time-Series Conditional GAN LSTM (TTSCGAN-LSTM), and the basic LSTM network. The performance of these models was analyzed for 168 items with various demand patterns ('Smooth', 'Erratic', 'Intermittent', 'Lumpy'). For performance evaluation, the data was divided into training, validation, and test datasets.

The training data spans from January 2019 to May 2022 and was used to learn demand patterns and develop models for future demand prediction. The validation data covers June 2022 to April 2023 and was used to assess the model's generalization ability and determine optimal hyperparameter combinations by checking for overfitting. Finally, the test data includes 12 weeks from May to August 2023, and the prediction results for this period were used to evaluate the actual performance of the models.

The performance evaluation of the models was conducted using three metrics: nRMSE (Normalized Root Mean Square Error), nMAE (Normalized Mean Absolute Error), and sMAPE (Symmetric Mean Absolute Percentage Error). Both nRMSE and nMAE quantify the prediction error of the models. nRMSE calculates the squared differences between predicted and actual values, averages them, takes the square root of this value, and normalizes it using the min-max method (the difference between the maximum and minimum values). nMAE calculates the average of the absolute differences between actual and predicted values and normalizes it in the same way. This normalization process makes it easier to compare the performance of different datasets or models, allowing the prediction accuracy of the models to be evaluated on a consistent basis regardless of range.

sMAPE calculates the absolute difference between actual and predicted values as a percentage of the sum of actual and predicted values, and then averages this value. The performance of the models was evaluated for each item, and the average performance for each metric across all items was calculated to represent the final model performance. This evaluation method helps to comprehensively assess the overall model performance while considering the performance differences for individual items.

In this study, we evaluated the demand forecasting performance of four models (MBB-LSTM, TCGAN-LSTM, TTSCGAN-LSTM, and LSTM) for 168 items with various demand patterns. The results showed that the modified models (MBB-LSTM, TCGAN-LSTM, TTSCGAN-LSTM) outperformed the standard LSTM model.

Table 4. smooth

Model	nRMSE	sMAPE	nMAE
MBB-LSTM	**0.246**	**0.076**	**0.184**
TCGAN-LSTM	0.286	0.090	0.220
TTSCGAN-LSTM	0.293	0.091	0.225
LSTM	0.307	0.100	0.241

Table 5. Erratic

Model	nRMSE	sMAPE	nMAE
MBB-LSTM	**0.322**	**0.165**	**0.241**
LSTM	0.396	0.173	0.307
TCGAN-LSTM	0.408	0.186	0.346
TTSCGAN-LSTM	0.438	0.191	0.368

Table 6. Intermittent

Model	nRMSE	sMAPE	nMAE
MBB-LSTM	**0.428**	0.355	**0.289**
TCGAN-LSTM	0.497	0.364	0.386
LSTM	0.505	0.373	0.434
TTSCGAN-	0.815	**0.327**	0.662

LSTM			

Table 7. Lumpy

Model	nRMSE	sMAPE	nMAE
MBB-LSTM	**0.357**	0.353	**0.284**
TCGAN-LSTM	0.517	0.287	0.454
TTSCGAN-LSTM	0.760	0.293	0.674
LSTM	0.842	**0.233**	0.724

For the 'Smooth' pattern, MBB-LSTM achieved the best performance with an nRMSE of 0.246, followed by the optimized TCGAN-LSTM and TTSCGAN-LSTM models, both with an nRMSE of 0.285. These results are an improvement over the standard LSTM model, which had an nRMSE of 0.307.

In the 'Erratic' pattern, MBB-LSTM also had the lowest error with an nRMSE of 0.322, compared to 0.396 for the LSTM, 0.408 for the TCGAN-LSTM model, and 0.438 for the TTSCGAN-LSTM model.

For the 'Intermittent' pattern, MBB-LSTM again showed the lowest error with an nRMSE of 0.428, followed by the TCGAN-LSTM model at 0.497 and the standard LSTM at 0.505. The TTSCGAN-LSTM model exhibited a relatively high error with an nRMSE of 0.815.

In the 'Lumpy' pattern, MBB-LSTM achieved the lowest error with an nRMSE of 0.357, followed by TCGAN-LSTM at 0.516, TTSCGAN-LSTM at 0.438, and the LSTM model with the highest error at 0.842. These results demonstrate that the modified models, particularly MBB-LSTM, consistently outperformed the standard LSTM model across various demand patterns.

The comparative analysis of prediction model performance for products with four different demand patterns revealed that models utilizing data augmentation techniques, such as MBB-LSTM and TCGAN-LSTM, exhibited superior performance, particularly for 'Lumpy' and 'Intermittent' patterns. These patterns are characterized by frequent periods of no or very low demand, making them difficult for traditional prediction models to capture. The enhanced performance of models with data augmentation in these challenging patterns can be attributed to their ability to generate diverse data, which helps in better capturing and reflecting the complex patterns within the data.

This study underscores the significant value of using data augmentation techniques in demand forecasting, especially when tailored to the specific demand patterns of products. The key findings are as follows. First, MBB-LSTM is highly effective for products with general demand patterns**,** delivering superior performance by accurately capturing regular trends and seasonality. Second, TCGAN-LSTM excels with products exhibiting special or complex demand patterns**,** such as 'Intermittent' and 'Lumpy', by effectively modelling irregularities and variability.

By enhancing the diversity and richness of the training data through appropriate augmentation strategies, these models achieve higher accuracy and reliability in their predictions. This highlights the importance of selecting and tailoring data augmentation techniques based on the specific characteristics of the data to improve demand forecasting performance.

7. Conclusion

This study introduces a novel methodology for enhancing demand forecasting accuracy by integrating data augmentation techniques with hybrid deep learning models. Through comprehensive experiments, we confirmed that models incorporating data augmentation—such as MBB-LSTM and TCGAN-LSTM—consistently outperform the baseline LSTM model across various demand patterns, including challenging ones like 'Lumpy' and 'Intermittent'.

This study makes the following academic contributions. First, it provides an innovative solution to the pervasive issue of data scarcity in demand forecasting. By employing time-series-specific data augmentation techniques, this study generates enriched and diverse datasets that enhance model training and generalization capabilities, even in data-constrained environments. This addresses a critical gap in the field where the applicability of machine learning models is often limited due to insufficient data.

Second, it presents a novel approach to handling high-variability patterns. Existing forecasting models often struggle to maintain consistent accuracy in scenarios with high variability and irregular demand, such as 'Lumpy' and 'Intermittent' patterns. The integration of tailored data augmentation techniques with hybrid deep learning models establishes a new standard for addressing these complex patterns, significantly improving prediction accuracy and robustness.

Third, this study advances the literature by proposing a scalable framework that effectively combines data augmentation with hybrid prediction models. This integration not only enhances forecasting performance but also broadens the applicability of demand forecasting techniques across various industries and dynamic scenarios.

Building on these academic contributions, this study also demonstrates practical utility for diverse industries. Most notably, it addresses data scarcity, enabling companies that previously struggled to adopt AI solutions to implement demand forecasting technologies. This is particularly impactful for small and medium-sized enterprises operating with limited or irregular data, thereby expanding the accessibility of AI-driven tools.

Additionally, the proposed models exhibit high predictive accuracy across various demand patterns, empowering businesses to optimize inventory levels, reduce costs, and prevent losses from stockouts. This ultimately enhances supply chain efficiency and maximizes operational performance. Furthermore, the framework's ability to maintain high accuracy in high-variability demand patterns allows businesses to better forecast customer needs and respond swiftly, thereby improving customer satisfaction and strengthening competitive advantage in the market.

This study was conducted using a specific dataset, which may limit the generalizability of the findings across different industries and demand conditions. To address this limitation, future research should employ a broader range of datasets that encompass various industry sectors and demand scenarios to validate the models' applicability and robustness.

Moreover, the effectiveness of data augmentation techniques is highly dependent on their alignment with the characteristics of the demand patterns. Inappropriate augmentation methods can degrade predictive performance. Therefore, in-depth research is necessary to systematically compare and evaluate different data augmentation techniques. Identifying the optimal methods tailored to specific demand patterns will help maximize forecasting accuracy and enhance model robustness.

Future studies should also explore new hybrid approaches that combine various data augmentation techniques with advanced prediction models. This exploration aims to enhance the

interpretability of the models and the reliability of the prediction results. Additionally, it is crucial to verify the predictive performance and practical applicability of these models in real-world settings to ensure their effectiveness in operational environments.

By addressing these limitations and pursuing the suggested avenues for future research, the full potential of integrating data augmentation with hybrid deep learning models in demand forecasting can be realized. This will contribute to developing more robust and accurate forecasting models, ultimately benefiting businesses through improved inventory management and supply chain optimization.

Acknowledgements

This research was supported by Ministry of SMEs and Startups of S.Korea, under grant number RS-2024-00441955.

Author contributions

Jinseop Yun: Conceptualization, Methodology, Software, Field study **Park Yejun:** Data curation, Writing-Original draft preparation, Validation., Field study **Doohee Chung:** Conceptualization, Writing-Reviewing and Editing.

Conflicts of interest

The authors declare no conflicts of interest.

References

[1] Bandara, K., Bergmeir, C., & Hewamalage, H. (2020). LSTM-MSNet: Leveraging forecasts on sets of related time series with multiple seasonal patterns. IEEE transactions on neural networks and learning systems, 32(4), 1586-1599.

[2] Bohanec, M., Borštnar, M. K., & Robnik-Šikonja, M. (2017). Explaining machine learning models in sales predictions. Expert Systems with Applications, 71, 416-428.

[3] Wheelwright, S., Makridakis, S., & Hyndman, R. J. (1998). Forecasting: methods and applications. John Wiley & Sons.

[4] Ramanathan, U. (2012). Supply chain collaboration for improved forecast accuracy of promotional sales. International Journal of Operations & Production Management, 32(6), 676-695.

[5] Seeger, M. W., Salinas, D., & Flunkert, V. (2016). Bayesian intermittent demand forecasting for large inventories. Advances in Neural Information Processing Systems, 29.

[6] Pinkus, A. (1999). Approximation theory of the MLP model in neural networks. Acta numerica, 8, 143-195.

[7] Elman, J. L. (1990). Finding structure in time. Cognitive science, 14(2), 179-211.

[8] Hong, J. K. (2021). LSTM-based Sales Forecasting Model. KSII Transactions on Internet & Information Systems, 15(4).

[9] Pliszczuk, D., Lesiak, P., Zuk, K., & Cieplak, T. (2021). Forecasting sales in the supply chain based on the LSTM network: the case of furniture industry.

[10] Salamanis, A., Xanthopoulou, G., Kehagias, D., & Tzovaras, D. (2022). LSTM-based deep learning models for long-term tourism demand forecasting. Electronics, 11(22), 3681.

[11] Sina, L. B., Secco, C. A., Blazevic, M., & Nazemi, K. (2023). Hybrid Forecasting Methods—A Systematic Review. Electronics, 12(9), 2019.

[12] Noh, J., Park, H. J., Kim, J. S., & Hwang, S. J. (2020). Gated recurrent unit with genetic algorithm for product demand forecasting in supply chain management. Mathematics, 8(4), 565.

[13] Makridakis, S., Spiliotis, E., & Assimakopoulos, V. (2018). The M4 Competition: Results, findings, conclusion and way forward. International Journal of Forecasting, 34(4), 802-808.

[14] Chen, Y., Xie, X., Pei, Z., Yi, W., Wang, C., Zhang, W., & Ji, Z. (2024). Development of a Time Series E-Commerce Sales Prediction Method for Short-Shelf-Life Products Using GRU-LightGBM. Applied Sciences, 14(2), 866.

[15] Schmidt, A., Kabir, M. W. U., & Hoque, M. T. (2022). Machine learning based restaurant sales forecasting. Machine Learning and Knowledge Extraction, 4(1), 105-130.

[16] Smirnov, P. S., & Sudakov, V. A. (2021, May). Forecasting new product demand using machine learning. In Journal of Physics: Conference Series (Vol. 1925, No. 1, p. 012033). IOP Publishing.

[17] Fourkiotis, K. P., & Tsadiras, A. (2024). Applying Machine Learning and Statistical Forecasting Methods for Enhancing Pharmaceutical Sales Predictions. Forecasting, 6(1), 170-186.

[18] The HR Director (2024). Dei initiatives on the rise, but strategic execution is lacking.

[19] Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning. Journal of big data, 6(1), 1-48.

[20] Wen, Q., Sun, L., Yang, F., Song, X., Gao, J., Wang, X., & Xu, H. (2020). Time series data augmentation for deep learning: A survey. arXiv preprint arXiv:2002.12478.

[21] Chen, M., Xu, Z., Zeng, A., & Xu, Q. (2023). FrAug: Frequency Domain Augmentation for Time Series Forecasting. arXiv preprint arXiv:2302.09292.

[22] Pouyanfar, S., Sadiq, S., Yan, Y., Tian, H., Tao, Y., Reyes, M. P., ... & Iyengar, S. S. (2018). A survey on deep learning: Algorithms, techniques, and applications. ACM Computing Surveys (CSUR), 51(5), 1-36.

[23] Gamboa, J. C. B. (2017). Deep learning for time-series analysis. arXiv preprint arXiv:1701.01887.

[24] Najafabadi, M. M., Villanustre, F., Khoshgoftaar, T. M., Seliya, N., Wald, R., & Muharemagic, E. (2015). Deep learning applications and challenges in big data analytics. Journal of big data, 2(1), 1-21.

[25] Zhang, T., Zhang, Y., Cao, W., Bian, J., Yi, X., Zheng, S., & Li, J. (2022). Less is more: Fast multivariate time series forecasting with light sampling-oriented mlp structures. arXiv preprint arXiv:2207.01186.

[26] Pascanu, R. (2013). On the difficulty of training recurrent neural networks. arXiv *preprint arXiv:1211.5063*.

[27] Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*.

[28] Che, Z., Purushotham, S., Cho, K., Sontag, D., & Liu, Y. (2018). Recurrent neural networks for multivariate time series with missing values. Scientific reports, 8(1), 6085.
[29] Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural computation, 9(8), 1735-1780.
[30] Abbasimehr, H., Shabani, M., & Yousefi, M. (2020). An optimized model using LSTM network for demand forecasting. Computers & industrial engineering, 143, 106435.
[31] Alzubaidi, L., Zhang, J., Humaidi, A. J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., ... & Farhan, L. (2021). Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions. Journal of big Data, 8, 1-74.
[32] Li, X., Xiong, H., Li, X., Wu, X., Zhang, X., Liu, J., ... & Dou, D. (2022). Interpretable deep learning: Interpretation, interpretability, trustworthiness, and beyond. Knowledge and Information Systems, 64(12), 3197-3234.
[33] Zhang, Y., Li, G., Muskat, B., & Law, R. (2021). Tourism demand forecasting: A decomposed deep learning approach. Journal of Travel Research, 60(5), 981-997.
[34] Kim, M., Choi, W., Jeon, Y., & Liu, L. (2019). A hybrid neural network model for power demand forecasting. Energies, 12(5), 931.
[35] Salman, S., & Liu, X. (2019). Overfitting mechanism and avoidance in deep neural networks. arXiv preprint arXiv:1901.06566.
[36] Giri, C., & Chen, Y. (2022). Deep learning for demand forecasting in the fashion and apparel retail industry. Forecasting, 4(2), 565-581.
[37] Alzubaidi, L., Bai, J., Al-Sabaawi, A., Santamaría, J., Albahri, A. S., Al-dabbagh, B. S. N., ... & Gu, Y. (2023). A survey on deep learning tools dealing with data scarcity: definitions, challenges, solutions, tips, and applications. Journal of Big Data, 10(1), 46.
[38] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... & Bengio, Y. (2014). Generative adversarial nets. Advances in neural information processing systems, 27.
[39] Bowles, C., Chen, L., Guerrero, R., Bentley, P., Gunn, R., Hammers, A., ... & Rueckert, D. (2018). Gan augmentation: Augmenting training data using generative adversarial networks. arXiv preprint arXiv:1810.10863.
[40] Han, Z., Zhao, J., Leung, H., Ma, K. F., & Wang, W. (2019). A review of deep learning models for time series prediction. IEEE Sensors Journal, 21(6), 7833-7848.
[41] Lee, S. W., & Kim, H. Y. (2020). Stock market forecasting with super-high dimensional time-series data using ConvLSTM, trend sampling, and specialized data augmentation. Expert systems with applications, 161, 113704.
[42] Iglesias, G., Talavera, E., González-Prieto, Á., Mozo, A., & Gómez-Canaval, S. (2023). Data Augmentation techniques in time series domain: a survey and taxonomy. Neural Computing and Applications, 35(14), 10123-10145.
[43] Bergmeir, C., Hyndman, R. J., & Benítez, J. M. (2016). Bagging exponential smoothing methods using STL decomposition and Box–Cox transformation. International journal of forecasting, 32(2), 303-312.
[44] Petitjean, F., Forestier, G., Webb, G. I., Nicholson, A. E., Chen, Y., & Keogh, E. (2014, December). Dynamic time warping averaging of time series allows faster and more accurate classification. In 2014 IEEE international conference on data mining (pp. 470-479). IEEE.
[45] Denaxas, E. A., Bandyopadhyay, R., Patiño-Echeverri, D., & Pitsianis, N. (2015, April). SynTiSe: A modified multi-regime MCMC approach for generation of wind power synthetic time series. In 2015 Annual IEEE Systems Conference (SysCon) Proceedings (pp. 668-674). IEEE.
[46] Kang, Y., Hyndman, R. J., & Li, F. (2020). GRATIS: GeneRAting TIme Series with diverse and controllable characteristics. Statistical Analysis and Data Mining: The ASA Data Science Journal, 13(4), 354-376.
[47] Kingma, D. P., & Welling, M. (2013). Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114.
[48] Yoon, J., Jarrett, D., & Van der Schaar, M. (2019). Time-series generative adversarial networks. Advances in neural information processing systems, 32.
[49] Bandara, K., Hewamalage, H., Liu, Y. H., Kang, Y., & Bergmeir, C. (2021). Improving the accuracy of global forecasting models using time series data augmentation. Pattern Recognition, 120, 108148.
[50] Iwana, B. K., & Uchida, S. (2021). An empirical survey of data augmentation for time series classification with neural networks. Plos one, 16(7), e0254841.
[51] Deng, Y., Liang, R., Wang, D., Li, A., & Xiao, F. (2023, November). Decomposition-based Data Augmentation for Time-series Building Load Data. In Proceedings of the 10th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation (pp. 51-60).
[52] Syntetos, A. A., Boylan, J. E., & Croston, J. D. (2005). On the categorization of demand patterns. Journal of the operational research society, 56, 495-503.
[53] Kim, J. S., Hwang, J. S., & Jung, J. W. (2020). A New LSTM Method Using Data Decomposition of Time Series for Forecasting the Demand of Aircraft Spare Parts. Korean Management Science Review, 37(2), 1-18.
[54] Yu, M., Tian, X., & Tao, Y. (2022). Dynamic Model Selection Based on Demand Pattern Classification in Retail Sales Forecasting. Mathematics, 10(17), 3179.
[55] Costantino, F., Di Gravio, G., Patriarca, R., & Petrella, L. (2018). Spare parts management for irregular demand items. Omega, 81, 57-66.
[56] Kuncoro, E. G. B., Aurachman, R., & Santosa, B. (2018, November). Inventory policy for relining roll spare parts to minimize total cost of inventory with periodic review (R, s, Q) and periodic review (R, S) (Case study: PT. Z). In IOP conference series: Materials science and engineering (Vol. 453, No. 1, p. 012021). IOP Publishing.
[57] Ramponi, G., Protopapas, P., Brambilla, M., & Janssen, R. (2018). T-cgan: Conditional generative

adversarial network for data augmentation in noisy time series with irregular sampling. arXiv preprint arXiv:1811.08295.
[58] Li, X., Ngu, A. H. H., & Metsis, V. (2022). Tts-cgan: A transformer time-series conditional gan for biosignal data augmentation. arXiv preprint arXiv:2206.13676.
[59] Efron, B. (1992). Bootstrap methods: another look at the jackknife. In Breakthroughs in statistics: Methodology and distribution, pages 569–593. Springer.
[60] Kunsch, H. R. (1989). The jackknife and the bootstrap for general stationary observations. The annals of Statistics, 1217-1241.
[61] Carlstein, E., Do, K. A., Hall, P., Hesterberg, T., & Künsch, H. R. (1998). Matched-block bootstrap for dependent data. Bernoulli, 305-328.
[62] Tian, X., Wang, H., & Erjiang, E. (2021). Forecasting intermittent demand for inventory management by retailers: A new approach. Journal of Retailing and Consumer Services, 62, 102662.