

Deep Learning based Seasonality and Trend Detection in Sales Forecasting

Tamilarasan Kannadasan¹

¹IT Industry Expert, Former Engineer at Meta, Amazon & Monster Worldwide, California, United States.

Article Info

Article history:

Received February 19, 2025

Revised April 10, 2025

Accepted April 19, 2025

Keywords:

Sales Prediction

Hybrid Deep Learning

Convolutional Neural Networks

Long Short-Term Memory

Time-Series Forecasting

ABSTRACT

Sales forecasting is essential for business planning, as it aids inventory management, marketing, and decision-making. Deep Learning combined with time-series analysis boosts prediction accuracy by capturing intricate temporal patterns. Precise sales forecasting remains difficult because of trends, seasonality, and noise. Previous techniques have issues with feature extraction and sequential dependencies, resulting in suboptimal efficiency. This study aims to develop a Hybrid Deep Learning (HDL) technique that combines the benefits of Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks to improve sales prediction accuracy. The primary emphasis is on combining feature extraction and temporal sequence learning to address the shortcomings of conventional methods. The proposed HDL framework prepares a sales dataset for time-series evaluation using a structured workflow that includes data exploration, preprocessing, and aggregation. To better comprehend the fundamental patterns, seasonal decomposition and autocorrelation analyses are used. The sliding window method is used to produce sequential data, which is then split into training and testing sets. Three predictive models—CNN, LSTM, and a hybrid CNN-LSTM—are built and trained using hyperparameter tuning. The models are evaluated using performance metrics such as root mean square error (RMSE), mean absolute error (MAE), and mean absolute percentage error (MAPE). Experimental results demonstrate that the proposed HDL surpasses CNN and LSTM with the lowest RMSE (2171.38), MAE (1219.79), and MAPE (538.18). The HDL technique combines CNN and LSTM to enhance sales prediction accuracy by capturing patterns and seasonality for better demand prediction and business evaluation.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author: Tamilarasan Kannadasan (e-mail: rktamil@gmail.com)

1. INTRODUCTION

Sales prediction is critical in inventory management, demand forecasting, and resource enhancement across industries [1]. Precise sales forecasting allows businesses to create informed decisions, effectively handle supply chains, and ensure product availability while avoiding overstocking [2]. With the development of machine learning and deep Learning, time-series forecasting methods have substantially enhanced prediction accuracy, providing new methods for evaluating and predicting sales data [3]. The increasing intricacy of customer behaviour, market patterns, and external factors like seasonality and promotional behaviours necessitates the utilization of sophisticated sales prediction models [4].

Conventional statistical models like AutoRegressive Integrated Moving Average (ARIMA), Exponential Smoothing, and linear regression models have all been used to forecast sales [5]. More recently, machine learning methods like Support Vector Machines (SVM), Random Forests (RF), and Gradient Boosting have been used to enhance prediction accuracy using historical sales data [6]. Deep learning models, especially Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM) networks, have demonstrated significant promise in capturing temporal dependencies and trends in time-series data. However, these models frequently struggle to extract intricate features from raw sales data, limiting their predicting capacity.

Despite the achievement of machine learning and deep learning models, current methods have numerous drawbacks. Conventional models, such as ARIMA, are only efficient for linear trends and frequently fail to capture the nonlinear relationships found in sales data. Additionally, while LSTM and RNN models are effective at sequence learning, they require a large amount of data and are computationally costly. These models also fail to recognize the significance of feature extraction from raw input data, which is critical for improving prediction accuracy. Furthermore, numerous current methods fail to fully incorporate both feature extraction and temporal dependency learning in a single cohesive framework, resulting in inadequate efficiency in intricate sales prediction tasks.

To tackle these drawbacks, this paper proposes a Hybrid Deep Learning (HDL) Approach for Sales Prediction that combines Convolutional Neural Networks (CNN) and LSTM networks. The CNN layers in the model extract intricate attributes from raw sales data, whereas the LSTM layers capture temporal dependencies, resulting in a more comprehensive comprehension of the data. This hybrid approach combines the advantages of both CNN and LSTM models, allowing for precise sales predictions even in the existence of nonlinear trends, seasonality, and noise.

The suggested HDL method starts with preprocessing the sales data, which includes normalization and sequence creation via a sliding window method. The CNN model extracts high-level features from temporal sales data, whereas the LSTM network deals with the sequential nature of the data by learning long-term dependencies. The ultimate predictions are generated by integrating CNN's feature extraction capacities with LSTM's temporal prediction capacities. The model is trained utilizing a rigorous training regimen, with hyperparameters tuned for accuracy. Unlike traditional techniques, our method incorporates feature extraction and sequential Learning in a unified framework, substantially enhancing prediction accuracy.

This paper provides the following contributions:

1. Introduction of a new HDL approach integrating CNN and LSTM networks for precise sales prediction.
2. An extensive methodology for sales data preprocessing, feature extraction, and temporal pattern learning.
3. The proposed model is empirically validated with practical sales data, showing greater effectiveness over conventional and previous deep learning models.
4. Detailed assessment of the model utilizing performance metrics like RMSE, MAE, and MAPE, with comparison to other models.

This research aims to propose and assess a new hybrid approach for sales forecasting utilizing deep learning methods. The objective is to design a model that integrates CNN and LSTM networks to extract pertinent features and capture temporal dependencies, thus enhancing the precision and robustness of sales predictions.

The novelty of this work lies in the incorporation of CNN with LSTM networks in a hybrid framework for sales forecasting. Unlike conventional models that concentrate solely on time-series prediction or feature extraction, this approach tackles both aspects concurrently, leading to more precise and dependable predictions.

The suggested HDL approach can be used in a variety of industries, such as retail, e-commerce, supply chain management, and inventory enhancement, where precise sales forecasting is critical to operational effectiveness. It can also be applied in the areas of demand prediction, financial market prediction, and production planning.

The organization of this paper is as follows: Section 2 offers an overview of existing techniques for sales prediction and time-series prediction, emphasizing both conventional techniques and current advances in machine and deep Learning. Section 3 provides a comprehensive explanation of the proposed HDL approach, including data preprocessing, model architecture, and training methods. Section 4 presents and analyzes the experimental findings, including a comparison of the HDL approach's effectiveness to previous models. Finally, Section 5 concludes the paper by analyzing the efficacy of the proposed method and suggesting directions for future study in sales prediction and hybrid deep learning methods.

2. LITERATURE REVIEW

Sales forecasting has emerged as an important research area for creating informed business decisions, especially in inventory management, demand prediction, and operational effectiveness. Several methodologies, ranging from classical statistical models to sophisticated machine learning methods, have been used to tackle the difficulties in this domain.

Suryawan et al. [7] examined the efficacy of ARIMA, LSTM, and Prophet techniques for predicting sales in the bakery industry. Their findings showed that ARIMA surpassed the other two techniques with the

smallest error metrics, containing a MAPE of 4.548%. This result emphasizes ARIMA's dependability for time-series data with linear patterns, rendering it a solid choice for companies with consistent sales patterns. In contrast, LSTM and Prophet demonstrated poor performance in capturing the nuances of bakery sales data, possibly because they relied on bigger datasets and more intricate trends.

Similarly, Hameed [8] examined the use of LSTM and ARIMA in retail sales prediction. The research emphasized the benefits of deep learning algorithms, such as LSTM, in capturing complex trends and dependencies in big datasets. While ARIMA excelled at forecasting stable patterns, LSTM showed its ability to handle nonlinear and nonstationary data. The study highlighted the importance of choosing suitable models depending on data characteristics and proposing hybrid solutions that combine statistical and deep learning techniques to improve prediction accuracy.

In contrast, Atmaja and Anandita [9] looked into single exponential smoothing (SES) for managing stocks in trading businesses. They showed how SES, an easier statistical technique, can accurately forecast varying sales patterns utilizing historical data. The research emphasized the significance of balancing inventory levels to meet consumer needs while reducing overstocking and deadstock. Despite its computational efficiency, SES's dependence on historical data and incapacity to capture intricate seasonality patterns may restrict its applicability for contemporary dynamic markets.

Building on this, Prasetyo et al. [10] proposed the Adaptive Response Rate Single Exponential Smoothing (ARRSES) approach for predicting sales in a microenterprise setting. ARRSES responded to changing data trends by dynamically adjusting alpha and beta parameters, attaining a MAPE of 9.85% and an accuracy of 90.15%. ARRSES's flexibility renders it appropriate for small businesses with varying sales volumes, though scalability for bigger datasets and intricate systems remains a difficulty.

Hong [11] presented an LSTM-based sales prediction model customized for clothing products that includes external variables such as temperature. The findings showed LSTM's capacity to recognize seasonal sales trends, like raised demand for shorts during the summer. This research emphasized LSTM's capability to incorporate time-series data with external factors, which improved its predictive accuracy for context-dependent sales prediction.

Pacella and Papadia [12] demonstrated LSTM's utility in supply chain management by using both forward and bidirectional LSTM models for short- and long-term demand prediction. Their results proved that LSTM is effective in handling nonlinear and nonstationary dynamics in intricate manufacturing systems, offering useful knowledge for replenishment planning and demand management.

Schmidt et al. [13] expanded the use of machine learning models, such as LSTMs and recurrent neural networks (RNNs), for restaurant sales prediction. Their findings demonstrated RNNs' better efficiency for datasets with trend and seasonality, with sMAPE scores below 20%. The study highlighted the importance of feature engineering and data preprocessing to improve model efficiency, especially for high-dimensional datasets.

Schmid et al. [14] conducted an empirical comparison of machine learning techniques, like XGBoost, and classical prediction methods for horticultural sales predictions. XGBoost consistently surpassed other techniques, particularly for datasets containing seasonal fluctuations and external variables such as weather. The research showed the benefits of integrating external meta-features in boosting algorithms, highlighting their scalability and resilience for practical uses.

Lakshmanan et al. [15] presented an LSTM-based model for forecasting sales demand utilizing historical data, which predicts product demand over three-time intervals. The data, gathered from southern Indian markets, was preprocessed and divided into 60% training and 40% testing, resulting in high precision and minimum errors. This research showed LSTM's capacity to accurately capture temporal trends in sales data.

Vallés-Pérez et al. [16] used RNNs and transformer architectures to predict sales at granular stages on the Corporación Favorita dataset. They developed a sequence-to-sequence model with minimum preprocessing and a new training technique to enhance time independence and generalization. With an RMSLE of 0.54, this study demonstrated the effectiveness of transformer models for precise sales forecasts.

Conventional sales prediction models, like ARIMA, are intended to capture linear patterns but struggle with the nonlinear trends and intricacies found in practical sales data. While deep learning models such as LSTM and RNN can manage sequential dependencies, they frequently need large datasets and substantial computational power. Furthermore, these models often overlook the crucial procedure of extracting features from raw data that can significantly enhance prediction accuracy. Previous techniques also tend to concentrate on either capturing temporal relationships or retrieving pertinent attributes individually, resulting in inadequacies in managing complex predicting tasks.

To tackle these gaps, this paper presents an HDL method that integrates Convolutional Neural Networks (CNN) for feature extraction and LSTM networks for temporal dependencies. By combining these two effective methods, the suggested approach seeks to offer a more extensive and effective strategy for sales

prediction, especially when dealing with nonlinear trends, seasonality, and noisy data. This hybrid approach promises to enhance forecast accuracy and resilience in intricate prediction situations.

3. METHOD

Sales prediction is essential for creating efficient business decisions because it allows organizations to forecast future sales patterns and assign resources more effectively. This research uses an HDL method to forecast sales values from time series data. To produce trustworthy and precise predictions, the methodology combines data preprocessing, feature extraction, and time-series modelling. This method guarantees a thorough evaluation of sales trends by integrating Convolutional Neural Networks (CNNs) for spatial feature extraction with LSTM networks for capturing temporal dependencies. The methodology also contains exploratory data analysis (EDA), seasonal analysis, autocorrelation analysis, and model evaluation, which offer comprehensive knowledge of the sales data.

3.1. Dataset Description

The dataset used in this research, known as the Superstore Sales Dataset, is sourced from Kaggle [17]. This dataset contains 9,800 rows and 18 attributes representing four years of retail sales data from a global superstore. The features contain consumer information (for example, Consumer Name, Segment), product information (for example, Product Name, Category), and sales metrics (for example, Order Date, Sales). The dataset's temporal nature renders it extremely appropriate for time-series analysis, allowing for sales trend prediction.

The dataset's context revolves around finding trends in retail sales and forecasting future sales using historical data. Time-series analysis can extract significant information from nonstationary data, which is critical for dynamic factors like economic patterns, seasonal variations, and consumer purchasing habits. The content contains features such as Order Date and Sales, which are essential for creating predictive models. This dataset was inspired by its application to practical business problems, which encourages data exploration and sales prediction for better decision-making.

3.2. Data Exploration

Exploratory Data Analysis (EDA) is a crucial step in comprehending the dataset's structure and getting it ready for analysis. The dataset is loaded into an appropriate data processing setting, and its structure is evaluated to guarantee it meets the necessities for time-series modelling. Temporal features, like Year and Month, are extracted from the Order Date column for trend analysis and aggregation. Sorting the dataset chronologically preserves the temporal order of sales data, which is required for precise time-series evaluation. This phase also identifies missing values and outliers to ensure data integrity before proceeding with further processing.

3.3. Data Preprocessing

Preprocessing is executed to improve the dataset and make it compatible with machine learning models. Irrelevant columns are eliminated, leaving only the most important features, like the order date, year, month, product name, and sales. The sales data is normalized utilizing Min-Max Normalization, which scales all values between 0 and 1 in Equation 1. This technique reduces the effect of extreme values and guarantees uniformity throughout the dataset.

$$\text{Normalized Value} = \frac{\text{Value} - \text{Min Value}}{\text{Max Value} - \text{Min Value}} \quad (1)$$

Where,

- Normalized Value: The result after executing normalization to the original data, scaled to a range between 0 to 1.
- Value: The original data point is to be normalized.
- Min Value: The smallest value in the dataset or the range being worked with.
- Max Value: The largest value in the dataset or the range being worked with.

Normalization is particularly helpful for neural network training since it accelerates convergence and improves model effectiveness by standardizing the input data. This formula is commonly used in data preprocessing to ensure that features contribute equally to the learning process, as defined by Han et al. [18].

3.4. Data Aggregation and Visualization

The data is combined by month and year to identify trends in sales performance over time. Aggregation offers a macro-level view of sales data, making it easier to detect seasonal patterns and long-term growth patterns. Furthermore, top-selling products are discovered by aggregating sales values by Product Name. Aggregated sales data is represented using visualization methods like bar charts (Figure 1) and line plots (Figure 2). These visualizations reveal seasonal spikes, dips, and other cyclical trends, offering a better comprehension of sales dynamics.

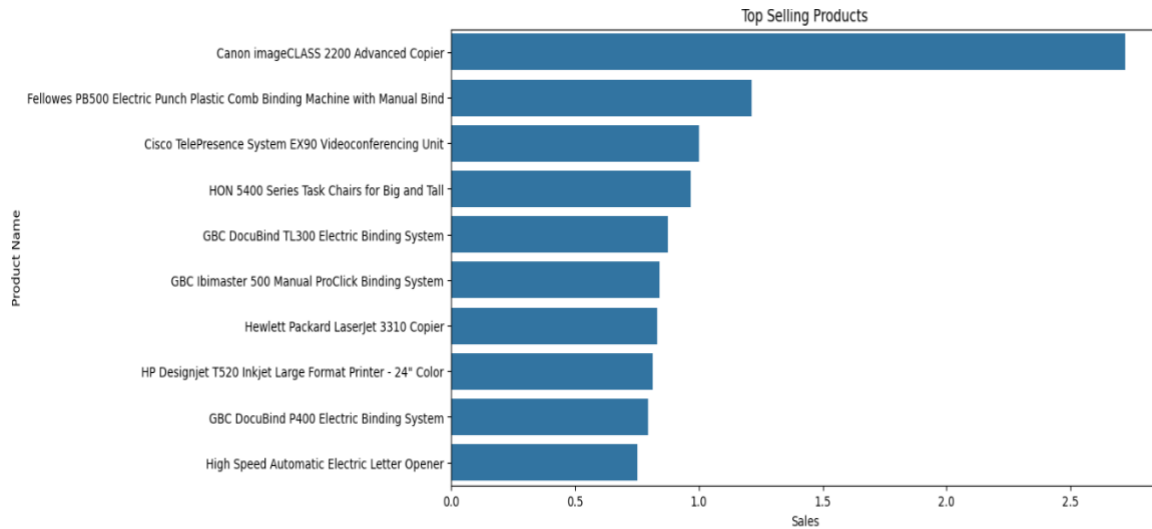


Figure 1. Bar chart of top-selling products, which are identified by summarizing sales values grouped by Product Name

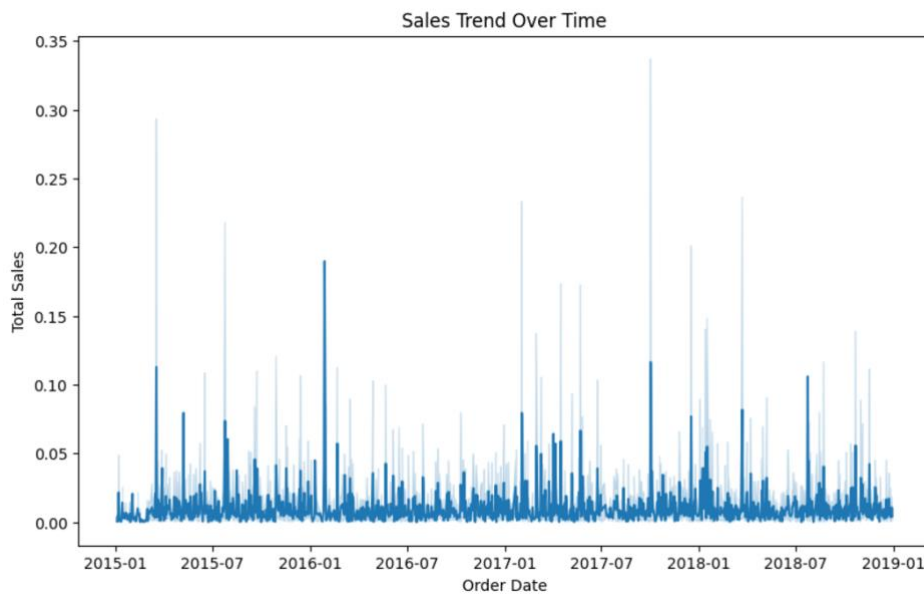


Figure 2. Line plot of Sales Trend Over Time

3.5. Seasonal Analysis

Seasonal analysis divides sales data into three elements: trend, seasonality, and residual. This decomposition aids in separating the long-term movement of sales values (trend), periodic fluctuations (seasonality), and random variations (residual). The decomposition can be depicted mathematically in equation 2.

$$Y_t = T_t + S_t + R_t \tag{2}$$

Where Y_t is the observed sales value at time t , T_t is the trend, S_t is the seasonality, and R_t is the residual. Comprehending these elements allows targeted predicting and decision-making. Figure 3 shows the seasonal breakdown of monthly sales data into three important components: trend, seasonality, and residuals. The trend depicts the long-term direction of sales over time, while seasonality captures regular fluctuations that occur year after year (like holiday peaks or off-season drops), and the residual represents random variations that are not explained by trend or seasonality. For instance, sales were significantly higher in December because of both a strong upward trend and seasonal demand, whereas June and November saw lower sales due to negative seasonality and residual factors. This breakdown aids in comprehending sales patterns and enhances the accuracy of predicting models.

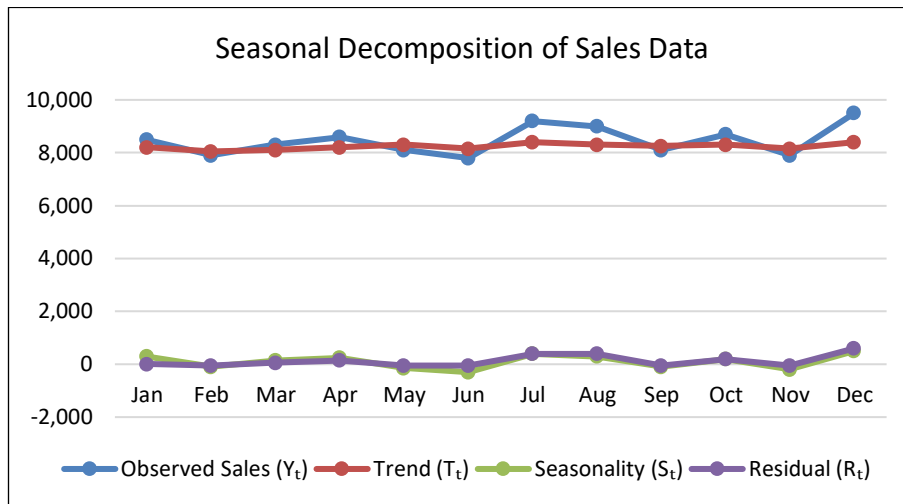


Figure 3. Seasonal Decomposition of Sales Data

3.6. Autocorrelation Analysis

Autocorrelation and partial autocorrelation analyses are used to investigate the relationships between current and historical sales figures. Autocorrelation measures how a time series value interacts with its lagged values. It is computed utilizing equation 3.

$$r_k = \frac{\sum_{t=1}^{n-k} (Y_t - \hat{Y})(Y_{t+k} - \hat{Y})}{\sum_{t=1}^n (Y_t - \hat{Y})^2} \quad (3)$$

Where,

r_k : The autocorrelation at lag k . It computes the correlation between the value at time t and the value at time $t+k$.

n : The total number of data points in the time series.

Y_t : The value of the time series at time t (for example, sales data at time t).

Y_{t+k} : The value of the time series at time $t+k$ (for example, the value at a time step shifted by k periods).

\hat{Y} : The mean (average) of all values in the time series. This value is subtracted from each Y_t and Y_{t+k} to centre the data, guaranteeing that the correlation concentrates on deviations from the mean rather than on absolute values.

These evaluations direct the choice of optimum lag values for time series modelling and feature engineering. Autocorrelation measures how current sales values compare to past values at various time lags, which aids in the identification of repeating patterns in time series data. The formula compares the similarity of values at time t and $t+k$, adjusting for the overall series average. This analysis informs the choice of appropriate lag features for time series models by emphasizing how strongly past sales influence future values. Figure 4 shows the Autocorrelation analysis.

3.7. Sequence Preparation

The dataset is restructured to generate sequential input-output pairs for time-series modelling, explained in equation 4. A sliding window method predicts sales value for the next day by using sequences of w days.

$$\text{Input: } [Y_t, Y_{t-1}, \dots, Y_{t-w+1}], \text{ Output: } Y_{t+1} \quad (4)$$

The sequence preparation step restructures the sales data utilizing a sliding window strategy, with each input containing sales values from the previous w days and the output representing the sales value for the following day. This method guarantees that the model learns from continuous patterns in the data, capturing the necessary time-based relationships for precise prediction.

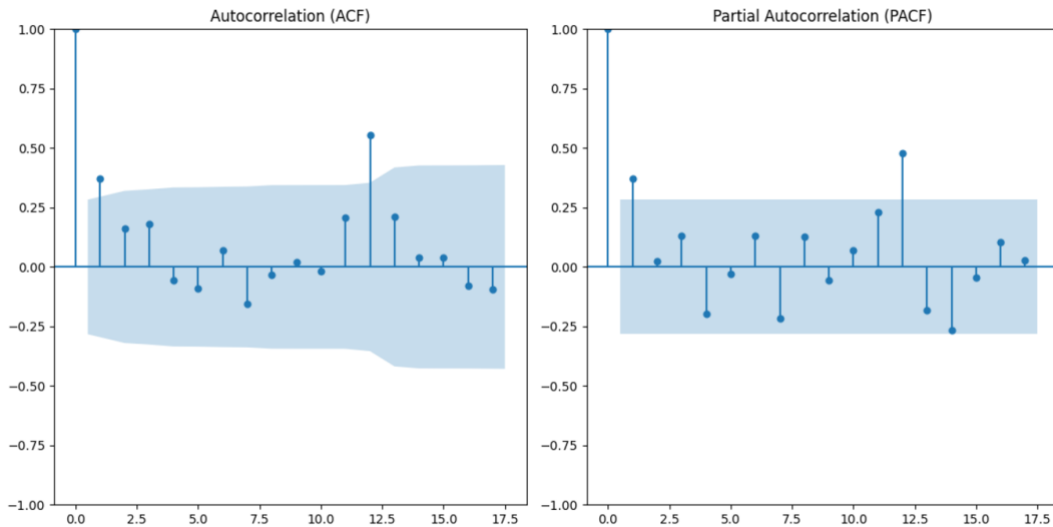


Figure 4. Autocorrelation analysis

3.8. Data Splitting

The dataset is divided into training and testing subsets in a 70:30 ratio to maintain the chronological order of sales data. The training set is employed to fit the models, and the testing set assesses their predictive efficiency.

3.9. Model Construction

3.9.1. CNN Model

The CNN employs convolutional layers to extract spatial patterns from sales sequences. The layers use a kernel of size k on the input data, followed by a ReLU activation function using equation 5.

$$\text{ReLU}(x) = \max(0, x) \quad (5)$$

Where,

x : The input to the ReLU (Rectified Linear Unit) function. This input could come from the output of a layer in a neural network after executing weights and biases.

ReLU(x): The output of the ReLU function. It is defined as the maximum of 0 and x , which means:

If $x > 0$, then ReLU(x)= x .

If $x \leq 0$, then ReLU(x)=0.

The CNN model employs convolutional layers to detect trends in sales sequences by applying filters of a specific size to the input data. Following the convolution operation, the ReLU activation function is employed, which returns the input value if positive and zero otherwise. This allows the model to learn complex patterns while remaining computationally effective and avoiding negative values that could impede Learning.

3.9.2. LSTM Model

The LSTM network uses sequential layers to capture temporal dependencies in sales data. The hidden state h_t at time t is calculated equation 6.

$$h_t = f(W_h h_{t-1} + W_x X_t + b) \quad (6)$$

Where,

h_t : The hidden state at time t , representing the result of the LSTM at that time step. It records pertinent data from the sequence up until time t .

f : The activation function, which is typically a nonlinear function like the tanh or sigmoid function, is performed to the weighted sum of the prior hidden state and the current input. It helps to introduce nonlinearity into the model.

W_h : The weight matrix related to the previous hidden state, which computes how much influence the prior hidden state h_{t-1} has on the current hidden state.

h_{t-1} : The hidden state at time $t-1$, which includes data from the prior time step that is passed along the sequence.

W_x : The weight matrix related to the current input X at t . This matrix computes how much influence the current input at time t has on the hidden state.

X_t : The input at time t , which could signify any feature or data point (for example, sales value, temperature, or other pertinent data).

b : The bias term is added to the weighted sum of the inputs and hidden state. It enables the network to shift the activation function and enhances model efficiency.

The LSTM model detects patterns in sales data over time by updating its hidden state at each time step using both the current input and the prior hidden state. This update is performed with a weighted sum of these values, a bias term, and a nonlinear activation function, such as tanh or sigmoid. The hidden state serves as the network's memory, enabling it to remember crucial data from earlier in the sequence and utilize it to make more accurate predictions.

3.9.3. Hybrid CNN-LSTM Model

The Hybrid CNN-LSTM model leverages the advantages of CNN and LSTM networks to manage both spatial and temporal elements of data. The CNN component is particularly effective at retrieving spatial features or patterns from sequential data, like sales patterns and variations over time. These attributes are then fed into the LSTM layers, which specialize in capturing long-term temporal dependencies and relationships over time steps. Lastly, the result from the LSTM layers is fed into fully linked dense layers, which conduct regression tasks to forecast the target sales figures. This hybrid approach guarantees that the model utilizes both spatial patterns and temporal dependencies, providing a more extensive and precise forecast for time series data.

3.10. Model Training

The training phase is critical in developing precise predictive models. In this research, each model—CNN, LSTM, and Hybrid CNN-LSTM—is trained with the specified parameters to enhance performance. The models are trained for a total of 150 epochs, enabling them to learn from data iteratively across numerous cycles. A batch size of 32 is employed, meaning the training data is separated into smaller subsets of 32 samples each, which are processed concurrently to enhance computational effectiveness.

The main objective of the training procedure is to reduce the error between actual and predicted sales values. This is accomplished by enhancing the loss function, which measures the error. The loss function utilized here is the Mean Squared Error (MSE), which is defined as equation 7.

$$Loss = \frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2 \quad (7)$$

Where,

n : The total number of data points or forecasts in the dataset or batch.

Y_i : The true value for the i -th data point or observation. For example, this could be the actual sales value for a given time period.

\hat{Y}_i : The predicted value for the i -th data point. It is the model's estimate of the true value.

$(Y_i - \hat{Y}_i)$: The variance (error) between the true value and the predicted value for the i -th data point.

$(Y_i - \hat{Y}_i)^2$: The squared error. Squaring the error penalizes higher discrepancies between the true and predicted values more severely.

Loss: The mean squared error (MSE) is the mean of all squared errors. It is frequently utilized as a loss function in regression issues, evaluating how well the model's predictions match the actual data.

The optimizer iteratively adjusts the model's weights to reduce this loss. The training procedure guarantees that the models gradually learn to enhance their forecasts by capturing both spatial and temporal trends in sales data.

To enhance learning effectiveness, the CNN, LSTM, and Hybrid CNN-LSTM models are trained over 150 cycles, known as epochs, with batches of 32 data samples at a time. The Mean Squared Error (MSE) loss function is used to reduce the difference between actual and predicted sales values by calculating the average of the squared differences between true and predicted values. A smaller MSE indicates higher prediction accuracy. During training, the model's weights are constantly updated to decrease this error, allowing the model to learn significant trends in sales data over time.

3.11. Prediction

After training, the models are tested on a dataset to forecast future sales values. The testing dataset is divided from the training data to guarantee that the models generalize well to previously unseen data rather than simply memorizing trends from the training set. The trained CNN, LSTM, and Hybrid CNN-LSTM models analyze the testing set's input sequences to create predicted sales values.

Because the sales data was normalized during preprocessing with Min-Max Normalization, the predicted values are initially on the normalized scale. To interpret these forecasts meaningfully, they are converted back to their original scale utilizing the inverse transformation of min-max normalization. The formula for this conversion is equation 8.

$$\text{Original Value} = \text{Normalized Value} \times (\text{Max Value} - \text{Min Value}) + \text{Min Value} \quad (8)$$

Where,

Normalized Value: The scaled prediction output from the model, which lies within the normalized range (e.g., 0 to 1).

Max Value: The maximum sales value in the original dataset, recorded before normalization.

Min Value: The minimum sales value in the original dataset, recorded before normalization.

This transformation guarantees that the predicted sales values are correctly scaled back to their original units, rendering them directly comparable to the actual sales values in the testing dataset. This stage is essential for assessing the models' efficacy and practical application in practical sales prediction. Algorithm 1 shows the HDL approach.

Algorithm 1: HDL approach

Input	: Superstore Sales Dataset (Order Date, Product Name, Sales, etc.) Sequence creation window size: 30 Model hyperparameters (CNN, LSTM, training settings)
Output	: Forecasted sales values
Step 1	: Data Preparation: <ul style="list-style-type: none"> • Load and examine the dataset. • Extract temporal features (Year, Month) and sort data sequentially. • Normalize sales values utilizing Min-Max Normalization.
Step 2	: Exploratory Analysis: <ul style="list-style-type: none"> • Combine sales by Month and Year to detect trends. • Analyze seasonality and trends utilizing visualizations and decomposition.
Step 3	: Sequence Generation: <ul style="list-style-type: none"> • Prepare input-output pairs for time-series modelling utilizing a sliding window. • Divide data into training (70%) and testing (30%) subsets.
Step 4	: Model Construction: <ul style="list-style-type: none"> • Construct CNN for spatial feature extraction. • Create LSTM for temporal pattern modelling. • Integrate CNN and LSTM in a hybrid model for improved predictions.
Step 5	: Model Training and Prediction: <ul style="list-style-type: none"> • Train models with specified hyperparameters. • Forecast sales on the testing set and reverse normalization for original values.

Furthermore, Figure 5 shows the flow diagram of the HDL approach. This methodology describes a reliable framework for sales prediction that combines time-series analytics and sophisticated neural network

architectures. By tackling both spatial and temporal dimensions of sales data, the HDL technique attains high accuracy and provides practical knowledge for decision-making.

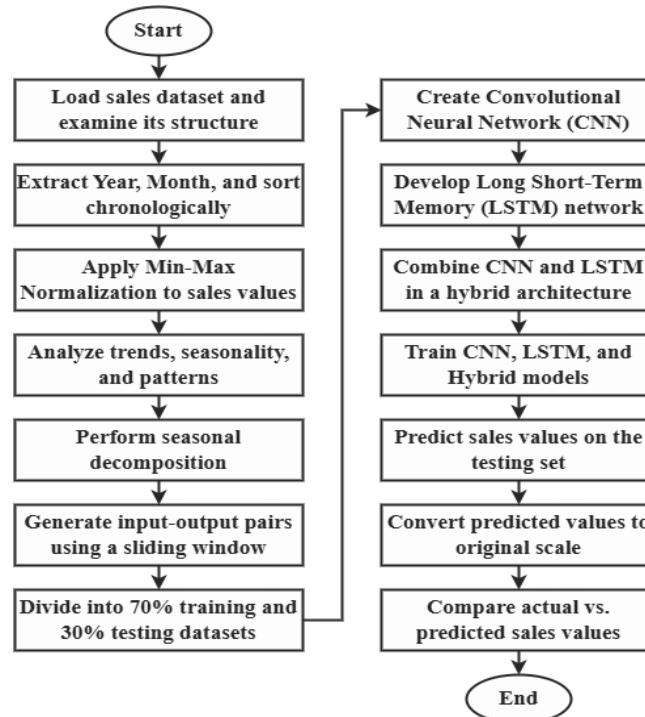


Figure 5. Flow diagram of HDL approach

4. RESULTS AND DISCUSSION

4.1. Experimental Setup

The sales prediction models were built in a cloud-based environment with Google Colab, an effective tool for coding and model training on GPUs to speed up computation. Google Colab facilitated the effective implementation of intricate functions such as data preprocessing, model training, and assessment without the requirement for expensive local computing resources. The CNN, LSTM, and HDL models were constructed, trained, and evaluated using machine learning and deep learning libraries like TensorFlow, Keras, and Scikit-learn. This environment enabled seamless collaboration and reproducibility of findings by accessing the same project from numerous devices.

4.2. Performance Metrics

The models were assessed utilizing three commonly employed performance metrics to evaluate the accuracy and efficiency of the sales forecasts.

Root Mean Square Error (RMSE): This metric measures the magnitude of prediction errors, with a higher penalty for large differences between predicted and actual values. It is sensitive to outliers, rendering it a useful metric when more errors require to be reduced for model optimization. The RMSE equation 9 explained below.

$$RMSE = \frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2 \quad (9)$$

Where,

n: The total number of data points or forecasts in the dataset or batch.

Y_i : The true value for the i -th data point or observation. For example, this could be the actual sales value for a given time period.

\hat{Y}_i : The predicted value for the i -th data point. It is the model's estimation of the true value.

Mean Absolute Error (MAE): MAE computes the mean of the absolute differences between predicted and actual values, resulting in a clearer and simpler picture of model effectiveness explained in equation 10. It is

less sensitive to outliers than RMSE, rendering it helpful when we require a simple measure of prediction accuracy.

$$MAE = \frac{1}{n} \sum_{i=1}^n |Y_i - \hat{Y}_i| \quad (10)$$

Mean Absolute Percentage Error (MAPE): MAPE expresses the prediction error as a percentage, which provides insight into the relative size of the errors using equation 11. It is particularly helpful when comparing models with various scales or when attempting to assess the model's efficiency in practical percentage terms.

$$MAPE = \frac{1}{n} \sum_{i=1}^n \left| \frac{Y_i - \hat{Y}_i}{Y_i} \right| \times 100 \quad (11)$$

4.3. Comparison Results

Table 1. Performance Comparison

Model	RMSE	MAE	MAPE
CNN	2967.65	2137.89	616.7
LSTM	2476.55	1720.23	628.68
HDL	2171.38	1219.79	538.18

As demonstrated in Table 1, the HDL model surpasses both CNN and LSTM models on all three evaluation metrics. The RMSE of HDL (2171.38) is lower than that of CNN (2967.65) and LSTM (2476.55), suggesting that the HDL model produces fewer and smaller errors on average. In terms of MAE, HDL scores 1219.79, which is substantially lower than CNN's 2137.89 and LSTM's 1720.23, indicating that the HDL model offers more precise predictions in terms of absolute differences from actual values. Finally, the MAPE of HDL (538.18) stands out as the lowest, indicating that the HDL model is the most effective in forecasting sales in relative percentage terms, rendering it highly precise and practical for predicting in different real-world applications.

4.4. Discussion

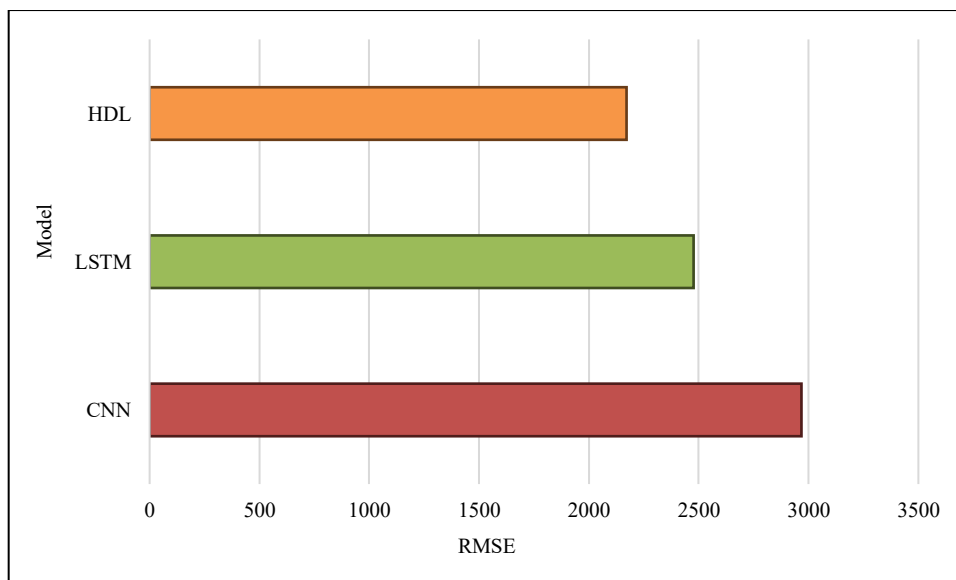


Figure 6. RMSE Comparison

In Figure 6, the HDL model has a lower RMSE value than CNN and LSTM, indicating that the hybrid architecture is better at reducing prediction errors, particularly larger errors. The CNN component of HDL effectively extracts spatial features from sales data, whereas the LSTM component captures temporal dependencies. This combination enables the HDL model to account for both short-term variations and long-

term trends in sales data, resulting in more dependable and precise predictions. As a result, the hybrid approach reduces RMSE, making it a more reliable model for sales forecasting.

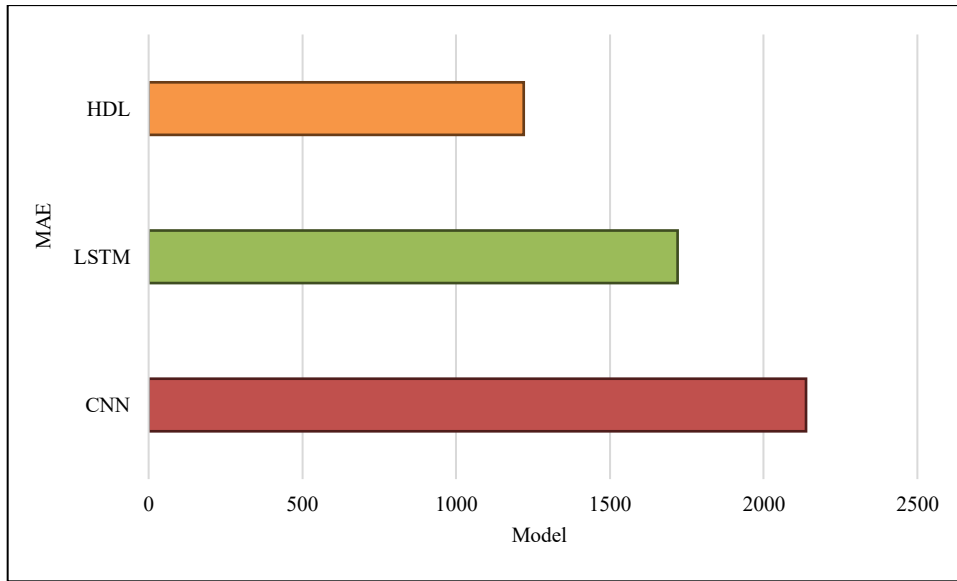


Figure 7. MAE Comparison

Figure 7 depicts the HDL model's better efficiency with the lowest MAE, demonstrating its capacity to decrease the average absolute deviation from actual sales values. The model effectively combines CNN's capacity to capture key spatial features with LSTM's managing of temporal patterns, resulting in highly accurate predictions. By decreasing MAE, HDL shows its potential for practical applications to minimize the absolute error in sales predictions, offering businesses more accurate predictions for better decision-making.

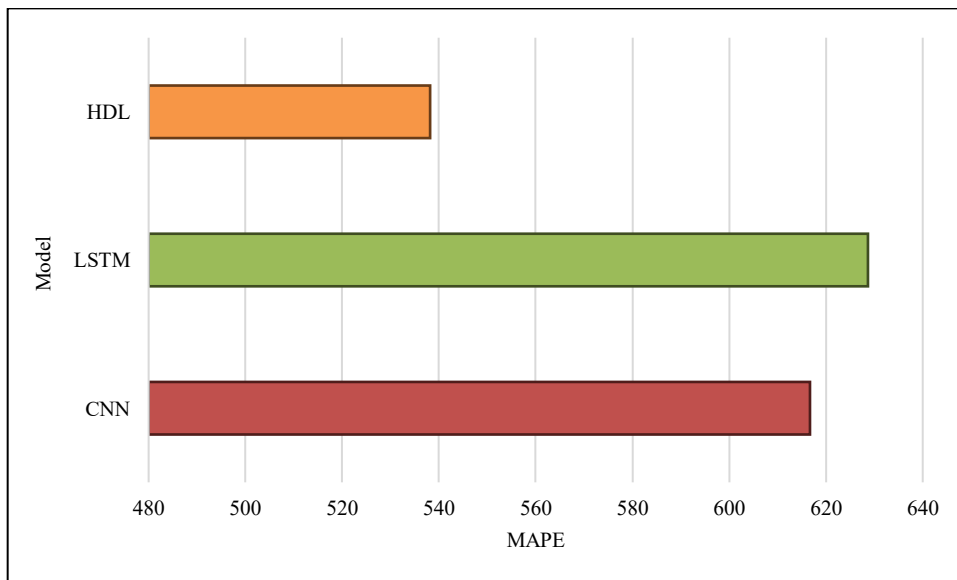


Figure 8. MAPE Comparison

In Figure 8, the HDL model has the smallest MAPE, indicating that it is better at reducing the relative percentage error when forecasting sales values. This metric is especially useful because it provides a percentage-based comparison, rendering it simpler to evaluate the model's predictive accuracy concerning the size of the actual sales values. The HDL model's hybrid architecture guarantees that both short-term and long-term dependencies are captured in the prediction procedure, resulting in a more dependable and consistent outcome in percentage terms than the CNN and LSTM models.

The experimental findings show that the HDL model is the most efficient method for sales forecasting among the three models tested. The HDL model surpasses CNN and LSTM in all three important

metrics—RMSE, MAE, and MAPE—demonstrating its better capacity to forecast sales with minimal error. The combination of CNN for spatial feature extraction and LSTM for temporal modelling produces a highly precise and resilient model that can comprehend both historical and seasonal patterns in sales data. This hybrid method proves to be an extremely effective strategy for sales prediction, with significant implications for business planning and decision-making.

5. CONCLUSION

In this research, an HDL model integrating CNN and LSTM was suggested for sales prediction. The experimental findings indicate that the proposed HDL model surpasses both CNN and LSTM models, attaining the lowest error rates with an RMSE of 2171.38, MAE of 1219.79, and MAPE of 538.18. The HDL model efficiently captures both spatial and temporal dependencies in sales data, resulting in precise predictions with minimal error. However, despite the promising outcomes, a few constraints remain, like the dependence on historical sales data for predictions, which may fail to capture sudden market changes or external factors. The excellence and completeness of the dataset may also influence the model's effectiveness. For future work, incorporating external variables like market patterns, promotional activities, and competitor behaviour may improve the model's predictive power. Furthermore, investigating more sophisticated hybrid architectures or transfer learning methods may aid in generalization across industries. Experimenting with various time-series prediction techniques and assessing the HDL model in practical scenarios will also be useful for evaluating its practical utility in dynamic business settings.

DATA AVAILABILITY STATEMENT

Data that support the findings of this study are available at <https://www.kaggle.com/datasets/rohitsahoo/sales-forecasting>

CONFLICTS OF INTEREST

The authors declare that they have no conflicts of interest in this work.

REFERENCES

- [1] F. Haselbeck, J. Killinger, K. Menrad, T. Hannus, and D. G. Grimm, "Machine Learning Outperforms Classical Forecasting on Horticultural Sales Predictions," *Mach. Learn. with Appl.*, vol. 7, p. 100239, Mar. 2022, doi: [10.1016/j.mlwa.2021.100239](https://doi.org/10.1016/j.mlwa.2021.100239).
- [2] M. Arunkumar, S. Palaniappan, R. Sujithra, and S. VijayPrakash, "Exploring Time Series Analysis Techniques for Sales Forecasting," 2024, pp. 41–55. doi: [10.1007/978-981-99-6755-1_4](https://doi.org/10.1007/978-981-99-6755-1_4).
- [3] V. Shah and S. Dimitrov, "A comparative study of univariate time-series methods for sales forecasting," *Int. J. Bus. Data Anal.*, vol. 2, no. 2, p. 187, 2022, doi: [10.1504/IJBDA.2022.126806](https://doi.org/10.1504/IJBDA.2022.126806).
- [4] S. Raizada and J. R. Saini, "Comparative Analysis of Supervised Machine Learning Techniques for Sales Forecasting," *Int. J. Adv. Comput. Sci. Appl.*, vol. 12, no. 11, 2021, doi: [10.14569/IJACSA.2021.0121112](https://doi.org/10.14569/IJACSA.2021.0121112).
- [5] M. Maslim, E. Ernawati, and K. Arinanda, "Motorcycle Parts Sales Forecasting Using Auto-Regressive Integrated Moving Average Model," *Int. J. Comput. Theory Eng.*, vol. 12, no. 1, pp. 28–31, 2020, doi: [10.7763/IJCTE.2020.V12.1259](https://doi.org/10.7763/IJCTE.2020.V12.1259).
- [6] Preethi Rajan, "Integrating IoT Analytics into Marketing Decision Making: A Smart Data-Driven Approach," *Int. J. Data Informatics Intell. Comput.*, vol. 3, no. 1, pp. 12–22, Feb. 2024, doi: [10.59461/ijdiic.v3i1.92](https://doi.org/10.59461/ijdiic.v3i1.92).
- [7] I. G. T. Suryawan, I. K. N. Putra, P. M. Meliana, and I. G. I. Sudipa, "Performance Comparison of ARIMA, LSTM, and Prophet Methods in Sales Forecasting," *sinkron*, vol. 8, no. 4, pp. 2410–2421, Oct. 2024, doi: [10.33395/sinkron.v8i4.14057](https://doi.org/10.33395/sinkron.v8i4.14057).
- [8] Y. Kaneko and K. Yada, "A Deep Learning Approach for the Prediction of Retail Store Sales," in *2016 IEEE 16th International Conference on Data Mining Workshops (ICDMW)*, IEEE, Dec. 2016, pp. 531–537. doi: [10.1109/ICDMW.2016.0082](https://doi.org/10.1109/ICDMW.2016.0082).
- [9] Ashish Kumar Pandey, "Analysis Role of ML and Big Data Play in Driving Digital Marketing's Paradigm Shift," *Int. J. Data Informatics Intell. Comput.*, vol. 2, no. 3, pp. 38–46, Sep. 2023, doi: [10.59461/ijdiic.v2i3.75](https://doi.org/10.59461/ijdiic.v2i3.75).
- [10] T. Arifin Prasetyo *et al.*, "Sales forecasting of marketing using adaptive response rate single exponential smoothing algorithm," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 31, no. 1, p. 423, Jul. 2023, doi: [10.11591/ijeecs.v31.i1.pp423-432](https://doi.org/10.11591/ijeecs.v31.i1.pp423-432).
- [11] Jun-Ki Hong, "LSTM-based Sales Forecasting Model," *KSII Trans. Internet Inf. Syst.*, vol. 15, no. 4, Apr. 2021, doi: [10.3837/tiis.2021.04.003](https://doi.org/10.3837/tiis.2021.04.003).
- [12] M. Pacella and G. Papadia, "Evaluation of deep learning with long short-term memory networks for time series forecasting in supply chain management," *Procedia CIRP*, vol. 99, pp. 604–609, 2021, doi: [10.1016/j.procir.2021.03.081](https://doi.org/10.1016/j.procir.2021.03.081).
- [13] A. Schmidt, M. W. U. Kabir, and M. T. Hoque, "Machine Learning Based Restaurant Sales Forecasting," *Mach. Learn. Knowl. Extr.*, vol. 4, no. 1, pp. 105–130, Jan. 2022, doi: [10.3390/make4010006](https://doi.org/10.3390/make4010006).
- [14] L. Schmid, M. Roidl, A. Kirchheim, and M. Pauly, "Comparing Statistical and Machine Learning Methods for Time Series Forecasting in Data-Driven Logistics—A Simulation Study," *Entropy*, vol. 27, no. 1, p. 25, Dec. 2024, doi: [10.3390/e27010025](https://doi.org/10.3390/e27010025).

- [15] B. Lakshmanan, P. S. N. Vivek Raja, and V. Kalathiappan, "Sales Demand Forecasting Using LSTM Network," 2020, pp. 125–132. doi: [10.1007/978-981-15-0199-9_11](https://doi.org/10.1007/978-981-15-0199-9_11).
- [16] I. Vallés-Pérez, E. Soria-Olivas, M. Martínez-Sober, A. J. Serrano-López, J. Gómez-Sanchís, and F. Mateo, "Approaching sales forecasting using recurrent neural networks and transformers," *Expert Syst. Appl.*, vol. 201, p. 116993, Sep. 2022, doi: [10.1016/j.eswa.2022.116993](https://doi.org/10.1016/j.eswa.2022.116993).
- [17] <https://www.kaggle.com/datasets/rohitsahoo/sales-forecasting>
- [18] P. J. Han J, Kamber M, "Data mining: concepts and techniques. 3rd ed," *Amsterdam: Morgan Kaufmann*, 2012.

BIOGRAPHIES OF AUTHORS



Tamilarasan Kannadasan received his Bachelor of Engineering degree from Anna University, Tamil Nadu, Chennai. He is a Sun Certified Java Professional and Web Component Developer with over 18 years of experience in software engineering. He has held prominent positions at globally recognized companies such as Meta Platforms, Amazon, and Monster Worldwide. He made significant contributions to the development of Meta's Creator Portal and Care Platform, which improved both user experience and operational efficiencies. Throughout his time at Amazon, he worked on the development of a managed network firewall service and system performance optimization. He has also demonstrated his technical expertise at hCentive Inc. and Computer Sciences Corporation, with a focus on application development and performance tuning. He is currently a Senior Member of IEEE and has served as a judge for the Globee Awards, demonstrating his respected position in the technology industry. His research interests include software engineering, system optimization, enterprise application development, and novel platform solutions. He can be contacted at email: rktamil@gmail.com.