Manoj et al. Iraqi Journal of Science, 2025, Vol. 66, No. 10, pp: 4502– 4516
DOI: 10.24996/ijs.2025.66.10.36

ISSN: 0067-2904

Enhancing Enterprise Performance Through Forecasting: A Deep RNN Approach

Aleesha Manoj[1], Ann Baby[1]*, A Francis Saviour Devaraj[2]*

[1]*Department of Computer Sciences, Rajagiri College of Social Sciences (Autonomous) Kalamassery - Kerala, India.*

[2]*University of Technology and Applied Sciences, Ibri, Sultanate of Oman.*

Received: 29/12/2023 Accepted: 2/10/2024 Published: 30/10/2025

Abstract

This research explores advanced methodologies in restaurant sales forecasting, focusing on a dataset from a Middle Eastern-based chain. It integrates machine learning algorithms and recurrent neural networks (RNN), including ARIMA, SARIMA, and LSTM models, to predict future sales trends accurately. Initial analysis includes various regression models to identify optimal sales prediction models, with Random Forest emerging as the top performer with an r2_score of 0.999895. The study emphasizes the importance of accurate sales forecasting in optimizing resource allocation, inventory management, and strategic decision-making for restaurant operations. Data preprocessing techniques such as missing value handling and feature selection ensure robust model performance. For time series forecasting, ARIMA and SARIMA models are applied to capture seasonal patterns, while LSTM models demonstrate ability in handling sequential data dependencies. The research employs comprehensive data integration strategies to unify diverse data sources into a cloud-based warehouse, enabling seamless analysis and forecasting. Results indicate that the selected models effectively predict restaurant sales, validating their applicability in real-world scenarios. The findings underscore the significance of machine learning and RNN models (LSTM) in enhancing sales prediction accuracy, thereby supporting informed business decisions in the competitive restaurant industry. Overall, this study contributes valuable insights into integrating advanced analytics for optimizing restaurant management and strategic planning.

Keywords: Sales forecasting, RNN, ARIMA, SARIMA, LSTM.

1. Introduction

Designing a sales forecast blends artistic insight with scientific rigor, vital for thriving businesses. Accurate predictions guarantee satisfied leadership, resilient enterprises, and motivated sales teams. The prime aim is to anticipate sales performance accurately, ideally surpassing expectations. This practice underpins sales operations, enabling judicious resource allocation, informed hiring, quota adjustment, and cost management. Sales forecasting enhances informed, impactful business choices. Restaurant sales forecasting manages inventory, staff deployment, and profit projections, profoundly impacting pivotal restaurant decisions. This research analyzes and forecasts restaurant sales, employing machine learning algorithms and RNN models. The outcome aids data-driven restaurant decisions, accentuating predictive analytics' value in steering future sales trends. Ultimately, this project effectively

*Email: saviodev@gmail.com

highlights machine learning and RNN models' effectiveness in an enterprise's sales prediction, delivering a valuable decision-making tool.

2. Scope and Objectives of the Research

Machine learning algorithms, including linear regression, ridge regression, lasso regression, decision trees, random forests, and XGBoost, are employed to pinpoint the optimal model for sales prediction. Following model selection, the DataFrame was fine-tuned by inputting specific food item and store names. This enabled a more focused sales analysis, unearthing potential trends. Using the selected model, predictions were generated using test data and then compared to actual sales figures. These results were visualized using plots, effectively illustrating any temporal changes or patterns. The research objectives investigated store sales across seasons, employing machine learning algorithms for modeling, filtering, and visualization. Lastly, leveraging recurrent neural network (RNN) models like ARIMA, SARIMA, and LSTM, sales forecasting for the forthcoming year was achieved, with LSTM's multivariate approach offering a robust insight into monthly restaurant sales for a futuristic period.

3. Research Methodology

The research methodology begins with the acquisition of raw data, followed by thorough preprocessing to convert it into a suitable format for subsequent analysis. This step ensures data quality and prepares it for detailed examination. The next phase involves exploratory data analysis (EDA), where the data is meticulously analyzed to identify underlying patterns, trends, and anomalies. This critical step helps in understanding the data's intrinsic properties and sets the stage for more advanced analysis.

Following EDA, the focus shifts to the identification of key features that have a significant impact on predictive outcomes. This process is crucial for enhancing the accuracy and efficiency of the predictive models. The data modeling phase then employs a variety of techniques, including linear regression, lasso regression, ridge regression, decision tree regression, random forest, and XGBoost, to generate predictions based on the existing dataset. These models are chosen for their robustness and ability to handle different data complexities.

For forecasting future trends, advanced time series models such as ARIMA, SARIMA, and LSTM are utilized. These models are adept at capturing temporal patterns and making accurate predictions about future data points. The final phase of the methodology involves the interpretation of the modeling results. Comprehensive reports are prepared to communicate the findings effectively. These reports summarize the results of the data analysis, feature identification, and modeling efforts, providing valuable insights and actionable recommendations based on the research. Figure 1 details the research schematic.

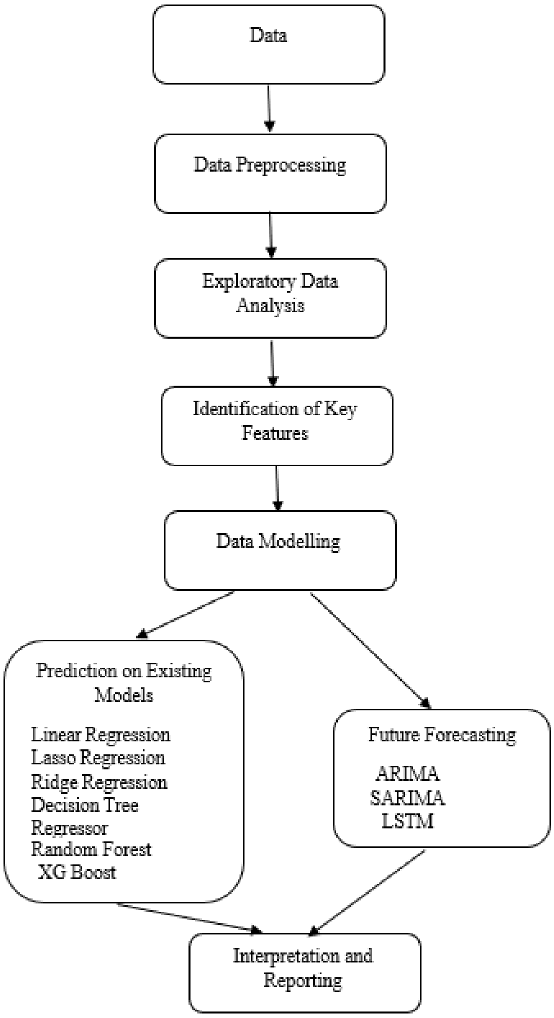

Figure 1: Schematic of the research

4. Related Work

Lasek et al. [1] propose machine learning-based approaches to predict restaurant sales by combining various regression models, including linear regression, random forest regression, and support vector regression. Pavlyshenko and Bohdan [2] propose a decision support system for restaurant sales forecasting using machine learning techniques. It compares multiple regression algorithms and demonstrates improved sales prediction accuracy. The study aims to enhance decision-making for restaurant managers, enabling better inventory management and resource allocation.

Research by Troise et al. [3] emphasizes that no single method universally outperforms others and highlights potential applications and future research directions in the foodservice industry. A comprehensive literature survey on restaurant sales and customer demand forecasting techniques categorizes the methods into seven groups, including time series analysis, regression analysis, and artificial neural networks. The study evaluates the advantages and disadvantages of each category and analyzes the performance of different forecasting techniques using real-world data from a restaurant chain in Canada.

Lutoslawski et al. [4] propose a machine learning approach for restaurant sales forecasting. It includes various models, such as random forest, linear regression, k-nearest neighbors, and trendy recurrent neural network (RNN) models. The study evaluates the performance of these models for one-day and one-week forecasting using real-world sales data from a midsized restaurant. RNN models exhibit strong performance for one-day forecasting, while simpler ML models excel in cases with trend and seasonality removed.

Ensafi et al. [5] predict restaurant sales using neural networks and consumer-based data collected from an online restaurant review platform. The authors compare the results with traditional time series models and find that the neural network model outperforms them in terms of accuracy. They conclude that consumer-based data and neural networks are effective tools for restaurant sales forecasting, benefiting decision-making for restaurant owners and managers. Novel approaches are developed to address limitations in machine learning methods for time series forecasting. They combine a recurrent neural network (RNN) with a dimension-reducing symbolic representation of the time series data. This combination aims to reduce sensitivity to hyperparameters and random weight initialization issues.

Elsworth et al. [6] detail that symbolic representation not only helps mitigate these problems but also potentially speeds up training without compromising forecast accuracy. The study demonstrates the effectiveness of this approach, highlighting its potential to enhance time series forecasting with improved training efficiency and performance.

Wisesa et al. [7] focus on sales prediction analysis for a telecommunications company using B2B sales data. The study employs intelligent data mining techniques and machine learning to create accurate predictive models for future sales trends. Big data challenges traditional forecasting methods, prompting the use of machine learning to improve prediction accuracy. The study evaluates various predictive models, with the gradient boost algorithm identified as the best performer. The research emphasizes the significance of reliable and accurate sales forecasts for the company's survival and growth in a competitive market. Intelligent data mining techniques and machine learning to create accurate predictive models for future sales trends. The dataset is preprocessed to extract relevant features and identify hidden patterns from the data. These extracted features will be used in designing an advanced version of a recurrent neural network (RNN) known as Long Short-Term Memory (LSTM). LSTM is chosen for its ability to handle sequential data and capture long-term dependencies effectively.

In [9], a distributed training approach is adopted by training the LSTM model efficiently, specifically following the data parallelism paradigm. This means that the training process involves multiple machines or processors working on different subsets of the data simultaneously. The Downpour training strategy is employed, which is a popular approach for distributed deep learning training. Chen et al., Boussaada et al., and Casado-Vara et al. [10], [11], [12] aim to develop a powerful AI model for better prediction and understanding of the underlying dynamics in the dataset by leveraging the power of distributed training and the capabilities of LSTM. The combination of LSTM and distributed training is expected to enhance the model's performance and scalability, making it suitable for large-scale and complex data analysis tasks.

5. Methodology and Experimental Details

5.1 Data Preparation

The research focuses on restaurant sales analysis and forecasting using a dataset collected from January 2018 to November 2022. The dataset originates from a Middle Eastern-based restaurant chain and encompasses a significant volume of data, comprising 1,047,556 instances and 16 attributes. The data is a compilation of information from various sources, providing a comprehensive view of different aspects of restaurant sales. With such a large dataset, the study aims to utilize advanced techniques, potentially including Long Short-Term Memory (LSTM) or other advanced recurrent neural networks, along with distributed training strategies to effectively analyze and forecast restaurant sales trends and patterns in the Middle East. The dataset includes the following attributes:

- store_id: Unique identifier for each store in the dataset.
- store_name: Name of the store.
- business_date: Date of the sales record.
- salescount: Number of sales transactions on the given business date.
- sales: Total sales amount on the given business date.
- itemprice: Price of the individual item sold.
- conversion_rate: Currency conversion rate on the given business date.
- id_global: Global identifier for the sales data.
- menu_name: Name of the menu or item sold.
- segment: Segment or category to which the menu item belongs.
- brands: Brand associated with the menu item.
- Branch_location: Location or branch of the store.
- country: Country where the sales data was recorded (e.g., Middle-East).
- currency: Currency used for sales transactions (e.g., Riyal, Dinar).
- grp_global_family: Group identifier for the sales data.
- major_group: Major group identifier for the sales data.

These attributes provide essential information about restaurant sales, including transaction details, menu items, pricing, currency, location, and groupings. The dataset's richness allows for comprehensive analysis and forecasting of restaurant sales trends and patterns in the Middle East. The research involved several steps to gather and integrate data from the customer's local database into the cloud-based data warehouse. The data integration process began with a comprehensive plan that identified source systems like Oracle Point of Sale, Material Control, and SAP B1, along with the specific data to be integrated. Prior to integration, a detailed analysis of stored data in each source system was conducted to understand its structure and relevance for the data warehouse. An appropriate extraction method was chosen based on the data warehouse's requirements, which included options such as direct database connections, APIs, or manual data entry. Integration points were configured by establishing database connections, APIs, or manual data entry forms.

After extraction, the gathered data underwent rigorous validation to ensure accuracy and completeness. Validated data was then loaded into the data warehouse using various methods, such as automated scripts, manual data entry, or direct database connections. Regular monitoring and auditing processes were implemented to continuously verify data accuracy and completeness, utilizing automated checks or manual inspections as needed. This systematic approach ensured that the integrated data was reliable and suitable for further analysis and business decision-making.

Following this comprehensive data integration plan, the business successfully collected and stored data from multiple source systems into the cloud-based data warehouse, providing a reliable and centralized repository for further analysis and decision-making.

5.2 Data Preprocessing

The following preprocessing was performed on the dataset to ensure accurate and effective data analysis:

- Missing values: The attributes *itemprice* and *menu_name* had 560 and 3021 missing values, respectively. To address this, the missing values in *itemprice* were replaced with the mean and the *menu_name* attribute was dropped.

- Date attribute: The *business_date* attribute was in object type and was split into *day, month,* and *year* for more accurate processing.
- Duplicate values: There were no duplicate values in the dataset, so no changes were made.
- Relevant columns: Irrelevant columns, such as *conversionrate, Country, currency, id_global,* and *year* were dropped. *conversionrate* and *Country* were not relevant because the dataset only contained data for Kuwait. *currency* was not useful in predicting *sales. id_global* was only for client comprehension.
- Label encoding: Label encoding was done prior to model building since most of the attributes were of object type.

6. Model Building

The research describes a machine learning model for sales data analysis and prediction. The data is divided into a training set comprising 7 months and a testing set comprising the remaining 4 months. Multiple machine learning models are utilized for the analysis and prediction of sales data. The models employed are linear regression, ridge regression, lasso regression, decision tree, random forest, and XGBoost. Each of these models is trained on the training data and evaluated based on their performance metrics. The goal is to determine which model provides the most accurate and reliable predictions for the sales data. The use of multiple models allows for a comprehensive comparison, allowing the best-performing model to be selected for the specific sales prediction task. By exploring different algorithms, the research aims to find the model that best captures the underlying patterns and relationships within the sales data.

Table 1: Model scores for various machine learning models

	Model	RMSE	r2_score
0	Linear Regression	49.609342	0.105715
1	Ridge	49.609334	0.105715
2	Lasso	49.440809	0.111781
3	Decision Tree Regressor	1.170416	0.999502
4	Random Forest	0.538504	0.999895
5	XGBoost	1.068176	0.999585

After evaluating the performance of each model from Table 1, Random Forest was identified as the best model based on its high r2_score of 0.999895, indicating a strong fit to the data, and its low root mean squared error (RMSE) of 0.538504, suggesting accurate predictions. Once we selected Random Forest as the best model, we proceeded to make predictions on the test data. This involved the following steps:

- Data Filtering: The DataFrame was filtered by specifying the name of the food item ('Weldone') and the store ('RE-Salhiya'). This focused the analysis on the specific sales data needed for prediction.
- Label Encoding: Categorical variables, such as food item names and store names, were encoded into numerical representations. This was necessary to feed the data into the machine learning model, which typically works with numerical inputs.
- Model Pipeline: A pipeline for the Random Forest model was created, which streamlines the process of data preprocessing, model fitting, and prediction.

Following these steps, the Random Forest model was effectively prepared to make predictions on the test data for the specified food item and store. The model pipeline ensured that the necessary transformations were applied to the data, and the label encoding facilitated the handling of categorical features. The predictions could be assessed by comparing them to the actual sales data from the test set, gauging the accuracy and efficacy of the Random Forest model, as illustrated in Figure 2.

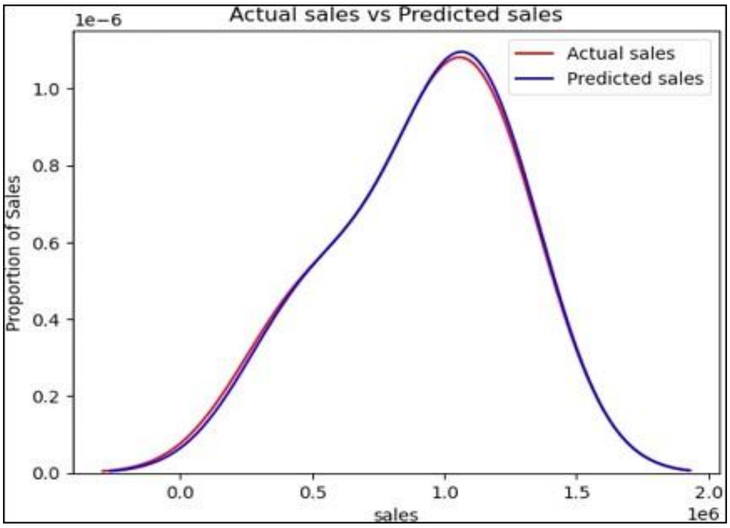

Figure 2: Visualization of actual sales vs predicted sales

The Random Forest model was used to analyze and predict the sales on the test data. This involved putting the test data into the model to generate predictions based on the model's learning from the training data, creating a model pipeline with StandardScaler() and Random Forest Regressor(), and fitting the pipeline with the training data. A data frame was created for each "month" by filtering the prediction for that month; the sales for each month were summed up, and a new dataframe was created with the actual sales and predicted sales values, which is shown in Figure 3.

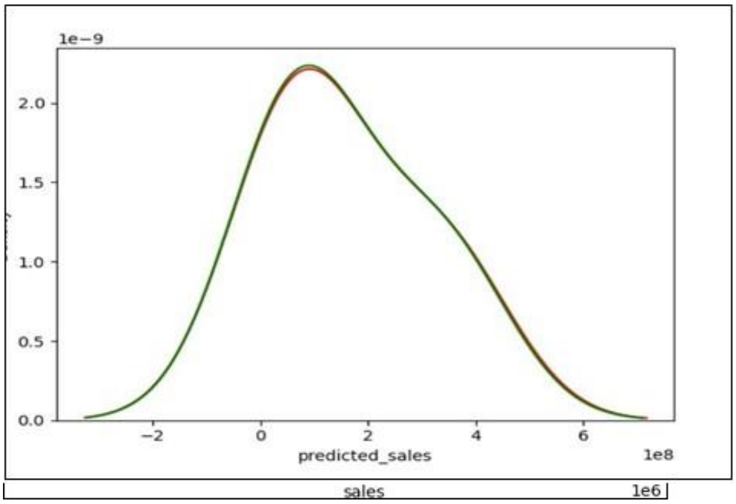

Figure 3: Model pipeline based actual sales vs predicted sales

By comparing the two lines, stakeholders and analysts can easily visualize how closely the predicted sales align with the actual sales for each month. A close alignment between the two lines indicates accurate predictions. This visual representation of the model's performance provides valuable insights into its accuracy and reliability. It allows decision-makers to quickly grasp the model's ability to forecast sales, which facilitates data-driven decision-making, resource allocation, and planning for the specific food item and store under consideration. Stakeholders and team members can easily understand and share the analysis results thanks to the plot, which serves as a powerful communication tool. Based on the visual plot comparing actual sales values with predicted sales values, it is evident that the Random Forest model's predictions closely align with the actual sales data. The closeness of the two lines on the plot indicates a high level of accuracy in the model's sales forecasts.

7. Analysis

The research aims to perform time series forecasting, predicting future sales based on historical sales data. [13], [14], [15], [16], [17] The models being used for this purpose are ARIMA, SARIMA, and LSTM. ARIMA (Auto Regressive Integrated Moving Average) and SARIMA (Seasonal ARIMA) are traditional time series models that consider the auto-regressive, moving average, and seasonal components of the data to make predictions. SARIMA (Seasonal Autoregressive Integrated Moving Average) is a predictive time series model that considers past observations and incorporates seasonal data patterns. A step beyond ARIMA, SARIMA introduces parameters to address seasonality. It identifies autoregressive, moving average, and seasonal aspects and synthesizes these for predictions. Long Short-Term Memory (LSTM) is a type of recurrent neural network (RNN) that is for both univariate and multivariate time series forecasting. LSTMs capture long-range dependencies in sequential data. According to research, the seasonal ARIMA model outperforms the traditional ARIMA model in forecast accuracy, particularly in capturing restaurant sales' seasonal patterns. Initially, only the sales data for the year 2022 was used for forecasting. However, recognizing the need for a longer historical context, the dataset was expanded to include sales data from the year 2018 onwards as well. This helps in providing a more comprehensive foundation for making future predictions. This research utilizes monthly sales data from 2018 to 2022. The target variable ('sales') is used for prediction in the univariate model.

7.1 ARIMA

The initial step in constructing an ARIMA model involves assessing the stationarity of the dataset. The Augmented Dickey-Fuller (ADF) test is used to determine the presence of a unit root in a time series. A p-value below 0.05 signifies stationary data, negating the need for differencing. When the ADF test is applied to the dataset, the resulting p-value of 0.010485 falls below 0.05, indicating pre-existing stationarity. Subsequently, the ARIMA model order was ascertained using two techniques:

7.1.1 AUTO ARIMA

This function automatically selects optimal parameters (p, d, q) for an ARIMA model, including seasonal components. The sales data was subjected to an AUTO ARIMA analysis, which identified the best model as ARIMA (0,1,0) (0,0,0) [0].

7.1.2 ACF and PACF Plot Analysis

The ARIMA model order can be deduced from the autocorrelation function (ACF) and partial autocorrelation function (PACF) plots, which are depicted in Figure 4 and Figure 5, respectively, in conjunction with AUTO ARIMA. ACF illustrates a series' correlation with past values, while PACF portrays correlation excluding prior lags' effects.

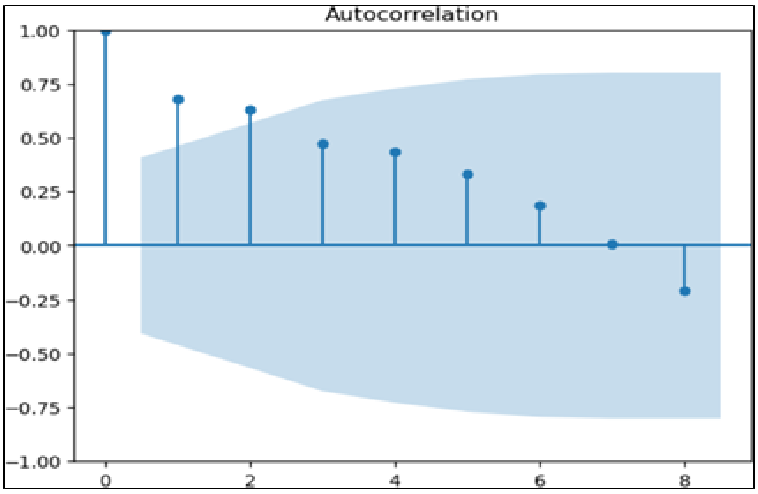

Figure 4: ACF Plot

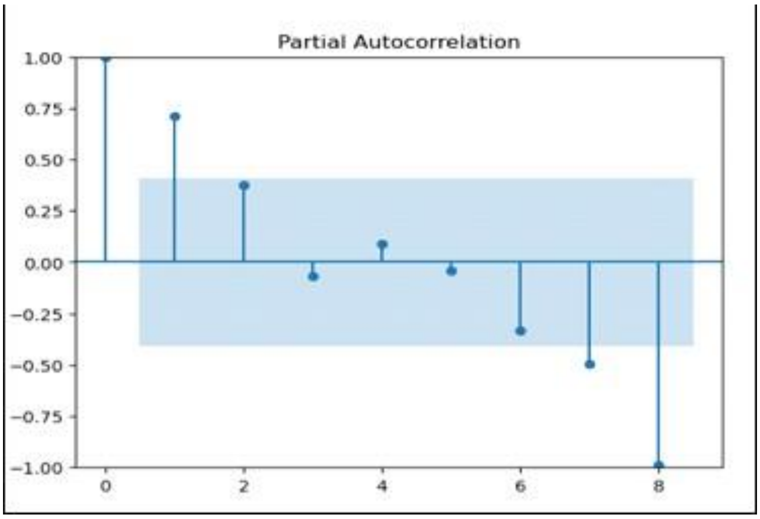

Figure 5: PACF Plot

Upon analyzing the autocorrelation function (ACF) and partial autocorrelation function (PACF) plots for the sales data, a distinct pattern emerged. The ACF plot exhibited a notable correlation with the first lag, diminishing rapidly afterward. Meanwhile, the PACF plot showed a significant correlation only with the first lag. These results led to the conclusion that the optimal ARIMA model order would be (1,0,0), indicating the use of one lag from the differenced series as a predictive component. The lowest Akaike Information Criterion (AIC) value, a metric comparing statistical model quality within a dataset, drove this choice.

After establishing the ARIMA model order, data scaling ensued, followed by partitioning into training and testing subsets. The training set encompassed 18 months, with the testing set spanning 5 months. Subsequently, an ARIMA model with an order of (1,0,0) was constructed using the training data. Model assessment was conducted using the summary function, furnishing insights into coefficients, standard errors, and statistical tests pertaining to the model's fit.

Consequently, the model's predictive capabilities were applied to forecast sales for the testing set. This was achieved by employing the forecast function with a research length of 5 months. To gauge the model's accuracy, the mean squared error (MSE) was computed using the predicted values in comparison to the actual values within the testing set. The resulting MSE value of 0.126 signifies the model's proficiency in generating precise sales predictions. The predicted sales values underwent a transformation to return to the original scale. This was achieved through the utilization of an inverse transform function, with the resultant values being stored within a designated data frame. This newly created data frame, housing the transformed predicted values, was merged with the original testing set. This merger resulted in the creation of a comprehensive data frame that included both the genuine sales values and the forecasted sales values derived from the ARIMA model. This amalgamated data frame served a dual purpose: it facilitated an easy comparison between the predicted and actual sales values while also enabling the visualization of the ARIMA model's prediction accuracy, as in Figure 6.

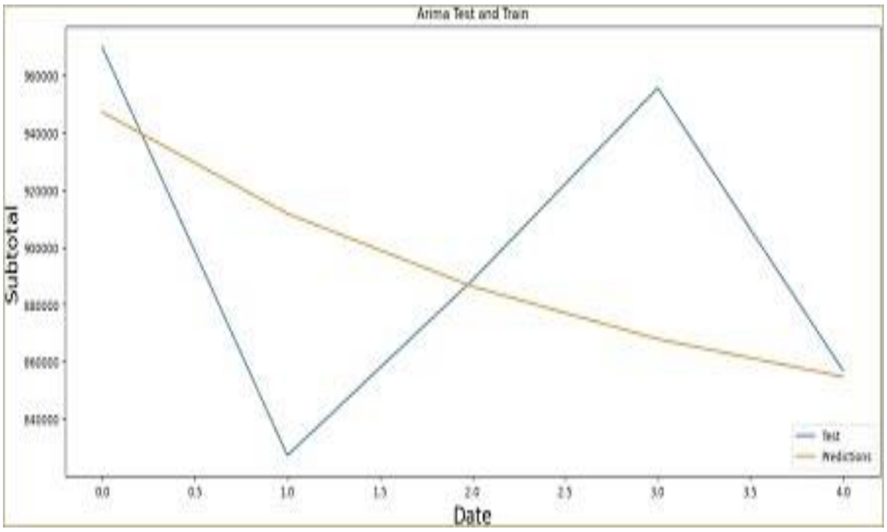

Figure 6: Actual ARIMA Plot

To predict sales for the upcoming 12 months, the ARIMA model was employed using the forecast function with a step parameter of 12. The ensuing forecasted values were transformed back to their original scale through the inverse transform function.

7.2 SARIMA

The SARIMA (Seasonal ARIMA) model was adopted because it recognized the ARIMA model's limitations in capturing data seasonality. The SARIMA process closely resembled the ARIMA process, but it differed by incorporating an additional parameter to represent seasonality. Following the same steps as ARIMA—scaling, splitting into training/testing, fitting, and forecasting—the SARIMA model integrated a parameter specifying the seasonality pattern in the data. With this enhancement, the SARIMA model adeptly captured seasonal trends, leading to heightened prediction precision. The employed SARIMA model boasted an order of (1,0,0) alongside a seasonal order of (1,0,0,4), denoting a recurring seasonal pattern every four months.

The training data underwent fitting of the SARIMA model, followed by generating predictions for the testing set. Assessment of the model's precision entailed computation of the root mean squared error (RMSE), resulting in a value of 0.13518778814692717. This observation signifies that the SARIMA model exhibited a slightly elevated error when contrasted with the ARIMA model.

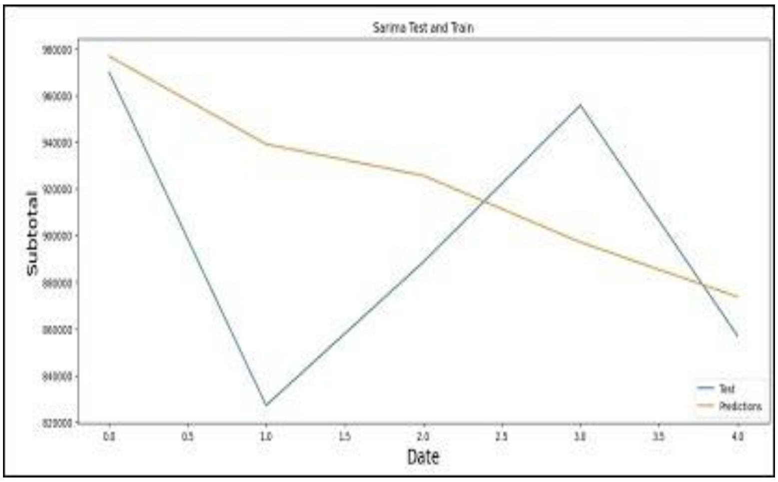

Figure 7 : Actual SARIMA Plot

Finally, the SARIMA model was harnessed to forecast sales for the forthcoming 12 months. Following prediction generation, the anticipated sales values were reverted to their original scale through inverse scaling. Subsequently, a DataFrame was created for the researched sales. This construction facilitated a clear presentation of sales predictions for the ensuing 12 months, while also considering the intricate interplay of seasonal patterns and trends within the data.

7.3 LSTM

This research paper employs recurrent neural networks (RNNs), specifically long short-term memory (LSTM) models, for time series forecasting of restaurant sales. LSTM models are chosen for their ability to capture temporal dependencies and handle sequences of data effectively. The paper utilizes both univariate and multivariate LSTM approaches to predict future sales trends based on historical sales data. These models are trained on a dataset spanning several years of restaurant sales data, enabling them to learn complex patterns and seasonal variations inherent in sales trends. The research evaluates the performance of LSTM models alongside traditional time series models like ARIMA and SARIMA, demonstrating LSTM's capability to provide accurate forecasts for monthly restaurant sales. In this context, the use of LSTM demonstrates its utility in enhancing predictive analytics and informing strategic decision-making for restaurant operations and resource management.

7.3.1 Univariate LSTM

For the implementation, numerous libraries were imported, including Keras, Sequential, LSTM, Dropout, Dense, and Time Series Generator. The dataset underwent conversion to datetime type, with the month column being designated as the index. Subsequently, data scaling was executed using the MinMaxScaler. Data was grouped into training and testing subsets, with generator instantiation for each segment. The model configuration encompassed four LSTM layers, each comprising 50 units, alongside a dropout rate of 0.2. The training process was executed utilizing the fit_generator method encompassing 10 epochs. Training was undertaken with the train generator, while validation was conducted via the validation generator. The model generated predictions for both the training and testing datasets. Evaluation of model performance involved computing the root mean squared error (RMSE) on the scaled testing data, yielding an outcome of 0.08.

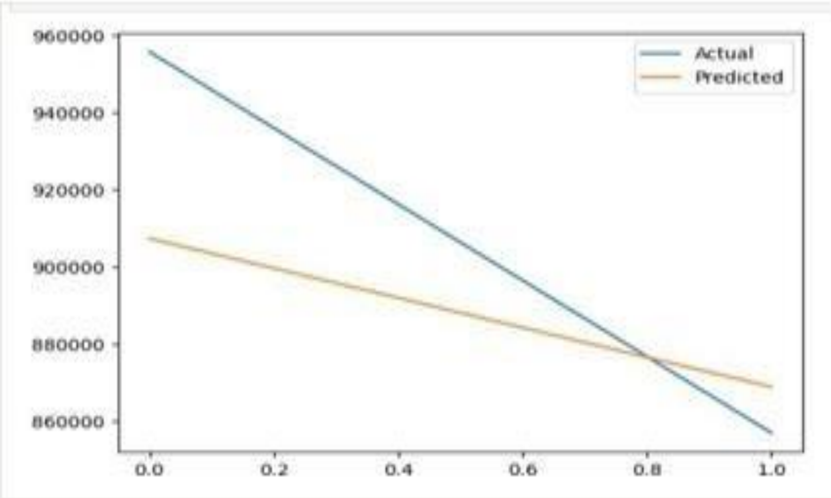

Figure 8: Univariate LSTM Plots

The scaling was inverted to get the actual sales values for the predictions made on the testing data. In the final stage, the univariate LSTM model embarks on forecasting the subsequent 12 sales values. This involves preparing the input data by extracting the last n_input values (3 in this context) from the testing data and reshaping it to conform to the LSTM model's input shape. A loop is then executed 12 times, yielding forecasted sales values for the ensuing 12 months. In each loop iteration, the model predicts the forthcoming sales value based on the previous input, appending it to the forecast_output list. Subsequently, forecast_input is updated: the first value is discarded, and the forecasted value is appended at the end. This modified forecast_input serves as input for the subsequent prediction. Ultimately, the forecasted sales values reside within the forecast_output list.

7.3.2 Multivariate LSTM

In the multivariate LSTM model, the 'month' column underwent transformation using sine and cosine functions to enhance forecasting sales. This transformation aimed to encapsulate the cyclic nature of monthly data. The newly generated 'month_sin' and 'month_cos' features encapsulate the 12-month cycle's cyclic patterns, enabling the model to learn seasonal highs and lows without imposing linear month- to-month assumptions. Such sine and cosine transformations are commonly embraced in time-series analysis, especially for cyclic or seasonal data. Furthermore, the 'season' column received one-hot encoding in the multivariate LSTM model. One-hot encoding transforms categorical data into numerical format suitable for machine learning. This approach involves converting the categorical 'season' column into four binary columns, each representing one of the four seasons. This encoding permits the model to differentiate between seasons and identify potential sales trends linked to specific seasons.

Data splitting followed an 80:20 train-test ratio, with input data scaled using the scikit-learn library's MinMaxScaler function. This scaling practice minimizes convergence challenges due to large input values in deep learning. The sequential class from Keras defined the LSTM network architecture. It encompassed four LSTM layers, each with decreasing neuron counts, and a dense output layer. After each LSTM layer, dropout layers intervened, counteracting overfitting. The commonly used sigmoid activation function was applied to all LSTM layers.

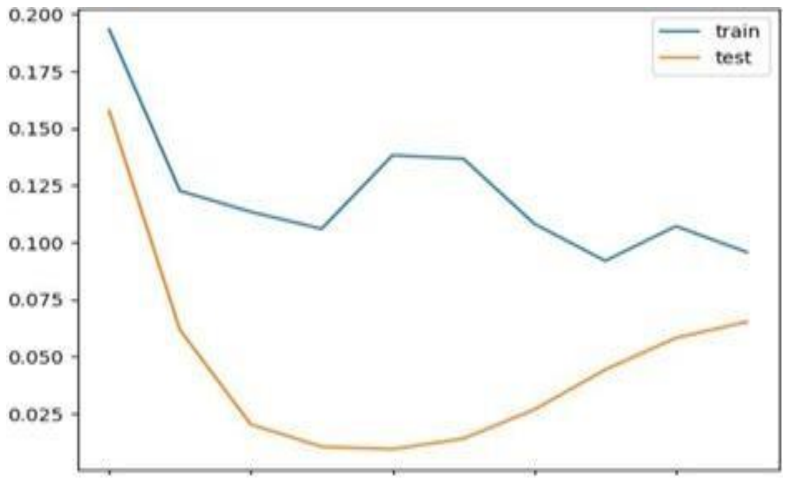

Figure 9: Multivariate LSTM Plots

TThe model was compiled using mean squared error (MSE) loss and the Adam optimizer (learning rate 0.275223). As regression problems don't require accuracy metrics, it was omitted. The model was trained on the training data using a batch size of 8, spanning 10 epochs. Validation data evaluated performance post-epoch. For potential learning curve analysis, the history object stores epoch-wise training and validation losses. Predicted sales values were rescaled using MinMaxScaler. Subsequent RMSE scores were computed for training and test sets. The predicted and actual sales values were printed for comparative assessment.

8. Findings and Results

Selecting the optimal model proved complex due to each model's distinct merits and drawbacks. For instance, while ARIMA disregarded seasonality, SARIMA was adopted to apprehend such cyclical patterns. Notably, ARIMA displayed an RMSE of 0.126211, while SARIMA and LSTM registered RMSE values of 0.13518 and 0.14, respectively. Several machine learning algorithms, including linear regression, ridge regression, lasso regression, decision trees, random forests, and XGBoost, were analyzed. Among these, Random Forest emerged as the optimal model, yielding an r2_score of 0.999895 and an RMSE of 0.538504.

Following model selection, particularly Random Forest, predictions were executed on the pre-divided test data. The dataset comprised a training subset spanning 7 months and a testing subset spanning 4 months. Impressively, the prediction outcomes exhibited remarkable promise, with predicted values closely mirroring the actual values. Each of the three models followed a similar approach: data division into training and testing subsets; model training using the training data; and subsequent testing on the testing data. Despite inherent differences in strengths and limitations among the models, a notable observation emerged that all three models yielded predictions remarkably proximate to the actual values.

Table 2 shows a comparison between the predicted ARIMA and SARIMA scores. Across all months, SARIMA generally predicts slightly higher sales figures compared to ARIMA. For instance, in the first month, SARIMA forecasts 976,827 units, while ARIMA predicts 947,356 units. With respect to trend comparisons over the months, both models show a consistent pattern in their forecasts. SARIMA tends to maintain a marginally higher forecast than ARIMA, indicating a potentially different modeling of seasonal effects or trends. In summary, while both ARIMA and SARIMA provide valuable insights into forecasting monthly restaurant sales, SARIMA shows a slight edge in predicting higher values consistently across the months in this particular dataset.

Table 2: Comparison of ARIMA and SARIMA Predictions.

Month	ARIMA	SARIMA
0	*947356.341463*	*976827.082633*
1	911798.120865	939123.364606
2	*886253.017267*	*925645.734137*
3	867901.358862	897070.929436
4	*854717.487250*	*873672.367340*
5	845246.164992	851052.290611
6	*838441.946327*	*831819.199540*
7	833553.780728	813205.943241
8	*830042.111849*	*796414.602272*
9	827619.321232	781022.563146
10	*825706.942924*	*767083.766635*
11	824404.926376	754305.921823

9. Future Work and Conclusion

There are several ways to improve the research. Firstly, including data spanning more than just the current period would offer insights that are more comprehensive. Secondly, exploring multivariate models incorporating factors like seasonal trends and holidays would boost prediction accuracy. Thirdly, extending the forecasting dataset to encompass at least four years would better capture seasonal behavior. Lastly, supplementing the dataset with attributes such as customer satisfaction and ratings would yield deeper sales trend insights. The research encompassed various exploratory data analyses, identifying top-selling attributes like branch, store, food, brand, and segment, while also scrutinizing sales patterns across months, seasons, and days. The objectives comprised evaluating food item sales in a specific store across months, analyzing store sales across seasons, and forecasting 12-month sales via algorithms like linear regression, decision tree, random forest, and XGBoost. Predicting future sales with ARIMA, SARIMA, and LSTM models resulted in successful attainment of research objectives. Future work could also include expanding the dataset to include more historical data and incorporating additional variables, such as customer feedback, for deeper insights into sales dynamics. Moreover, the integration of advanced analytics in restaurant management will facilitate data-driven decision-making, leading to increased efficiency, reduced costs, and enhanced customer satisfaction. The insights gained from this research can be applied across various verticals of restaurant operations, increasing overall business growth and competitiveness.

References

[1] Lasek, Agnieszka, Nick Cercone, and Jim Saunders. "Restaurant sales and customer demand forecasting: Literature survey and categorization of methods." *Smart City 360°: First EAI International Summit, Smart City 360°, Bratislava, Slovakia and Toronto, Canada,. Revised Selected Papers,* vol 1, pp. 479-491, October 2017.

[2] Pavlyshenko, Bohdan M. "Machine-learning models for sales time series forecasting." *Data* 4, Vol.1 -15, 2019, doi: 10.3390/data4010015.

[3] C. Troise, A. O'Driscoll, M. Tani, and A. Prisco, "Online food delivery services and behavioural intention–a test of an integrated TAM and TPB framework." *British Food Journal* .2, vol. 123, no. 2, pp. 664– 683, 2020, doi: 10.1108/BFJ-05-2020-0418.

[4] K. Lutoslawski, M. Hernes, J. Radomska, M. Hajdas, E. Walaszczyk, and A. Kozina, "Food Demand Prediction Using the Nonlinear Autoregressive Exogenous Neural Network*," IEEE Access*, vol. 9, pp. 146123–146136, 2021, doi: 10.1109/ACCESS.2021.3123255.

[5] Ensafi, Yasaman, Saman Hassanzadeh Amin, Guoqing Zhang, and Bharat Shah, "Time-series forecasting of seasonal items sales using machine learning – A comparative analysis," *International Journal of Information Management Data Insights*, vol. 2, no. 1, p. 100058, 2022, doi: 10.1016/j.jjimei.2022.100058.

[6] Elsworth, Steven, and Stefan Güttel. "Time series forecasting using LSTM networks: A symbolic approach." *arXiv preprint arXiv:2003.05672* ,2020. Available: http://arxiv.org/abs/2003.05672.

[7] Wisesa, Oryza, Andi Adriansyah, and Osamah Ibrahim Khalaf, "Prediction Analysis for Business To Business (B2B) Sales of Telecommunication Services using Machine Learning Techniques," *Majlesi Journal of Electrical Engineering,* vol. 14, no. 4, pp. 145–153, 2020, doi: 10.29252/mjee.14.4.145.

[8] Lim, Bryan, and Stefan Zohren. "Time-series forecasting with deep learning: a survey." *Philosophical Transactions of the Royal Society A,* vol. 379, no. 2194, 2021, doi: 10.1098/rsta.2020.0209.

[9] R. Casado-Vara, A. M. del Rey, D. Pérez-Palau, L. De-La-fuente- valentín, and J. M. Corchado, "Article web traffic time series forecasting using LSTM neural networks with distributed asynchronous training," *Mathematics*, vol. 9, no. 4, pp. 1–22, 2021, doi: 10.3390/math9040421.

[10] K. Chen, S. Laghrouche, and A. Djerdir, "Prognosis of fuel cell degradation under different applications using wavelet analysis and nonlinear autoregressive exogenous neural network," Renewable Energy, vol. 179, pp. 802–814, 2021, doi: 10.1016/j.renene.2021.07.097.

[11] Z. Boussaada, O. Curea, A. Remaci, H. Camblong, and N. M. Bellaaj, "A nonlinear autoregressive exogenous (NARX) neural network model for the prediction of the daily direct solar radiation," *Energies*, vol. 11, no. 3, 2018, doi: 10.3390/en11030620.

[12] Casado-Vara, R., Martin del Rey, A., Pérez-Palau, D., de-la-Fuente-Valentín, L., & Corchado, J. M. "Web traffic time series forecasting using LSTM neural networks with distributed asynchronous training", *Mathematics*, vol 9. No. 4, p. 421, 2021.

[13] Sah, S., Surendiran, B., Dhanalakshmi, R., Mohanty, S. N., Alenezi, F., & Polat, K., Forecasting COVID-19 pandemic using Prophet, ARIMA, and hybrid stacked LSTM-GRU models in India. *Computational and Mathematical Methods in Medicine,* vol. 1, 2022, doi: 10.1155/2022/1556025.

[14] T. Bashir, C. Haoyong, M. F. Tahir, and Z. Liqiang, "Short term electricity load forecasting using hybrid prophet-LSTM model optimized by BPNN," *Energy Reports,* vol. 8, pp. 1678–1686, 2022, 2022, doi: 10.1016/j.egyr.2021.12.067.

[15] K. E. ArunKumar, D. V. Kalaga, C. Mohan Sai Kumar, M. Kawaji, and T. M. Brenza, "Comparative analysis of Gated Recurrent Units (GRU), long Short-Term memory (LSTM) cells, autoregressive Integrated moving average (ARIMA), seasonal autoregressive Integrated moving average (SARIMA) for forecasting COVID-19 trends," *Alexandria Engineering Journal.,* vol. 61, no. 10, pp. 7585–7603, 2022, doi: 10.1016/j.aej.2022.01.011.

[16] T. Falatouri, F. Darbanian, P. Brandtner, and C. Udokwu, "Predictive Analytics for Demand Forecasting - A Comparison of SARIMA and LSTM in Retail SCM," *Procedia Computer Science,* vol. 200, no. 2019, pp. 993–1003, 2022, doi: 10.1016/j.procs.2022.01.298.

[17] Y. Yang, C. J. Fan, and H. L. Xiong, "A novel general-purpose hybrid model for time series forecasting," *Applied Intelligence,* vol. 52, no. 2, pp. 2212–2223, 2022, doi: 10.1007/s10489-021-02442-y.