IJMSERH

International Journal of Multidisciplinary and Scientific Emerging ResearcH (IJMSERH)

Volume 14, Issue 2, April-June 2026

Impact Factor: 9.274

ROAD INNO SPACE SJIF Scientific Journal Impact Factor doi Crossref

editor@ijmserh.com www.ijmserh.com

International Journal of Multidisciplinary and Scientific Emerging ResearcH (IJMSERH)

| ISSN: 2349-6037 | www.ijmserh.com ||Impact Factor: 9.274||Quarterly, Peer Reviewed & Refereed Journal|

|| Volume 14, Issue 2, April-June 2026 ||

DOI: 10.15662/IJMSERH.2026.1402028

Predictive Analytics for Sales Forecasting Using Python

K. Rajeswari, D. Saroj Priyadharsini

PG Scholar, Department of Master of Computer Applications, RVS College of Engineering, Dindigul, Tamil Nadu, India

Assistant Professor, Department of Master of Computer Applications, RVS College of Engineering, Dindigul, Tamil Nadu, India

ABSTRACT: Sales forecasting is an important process that helps businesses predict future demand, manage inventory, improve financial planning, and develop effective marketing strategies. Traditional forecasting methods often fail to handle large datasets, seasonal trends, and changing customer behaviour, resulting in inaccurate predictions and operational inefficiencies. To overcome these challenges, this project proposes a Predictive Analytics for Sales Forecasting system using Python and machine learning techniques. The system analyses historical sales data and customer purchasing patterns using algorithms such as Linear Regression, Random Forest, ARIMA, and LSTM. Various processes, including data preprocessing, feature engineering, model training, and performance evaluation, are implemented to improve forecasting accuracy. Python libraries such as Pandas, NumPy, Scikit-learn, TensorFlow, and Matplotlib are used for data analysis and model development. The proposed system also provides graphical visualisation of sales trends and forecasting results, helping businesses better understand customer demand patterns and market behaviour. Advanced predictive models improve forecasting efficiency by identifying seasonal variations and hidden trends in sales data. The system supports better resource allocation, production planning, and strategic business decision-making. The experimental results show that the proposed system provides more accurate and reliable sales predictions compared to traditional forecasting methods, helping organisations optimise inventory management, improve business planning, and support data-driven decision-making.

KEYWORDS: Sales Forecasting, Predictive Analytics, Machine Learning, Python, LSTM, ARIMA.

I. INTRODUCTION

In the modern era of digital transformation, businesses are increasingly relying on data-driven approaches to remain competitive and responsive to market demands. Sales forecasting is one of the most critical components of business analytics, as it directly influences decision-making in areas such as inventory control, production planning, financial budgeting, and customer relationship management. Accurate forecasting enables organizations to anticipate demand fluctuations, allocate resources efficiently, and maintain a balance between supply and demand.

However, traditional forecasting methods often lack the capability to process high-dimensional and large-scale datasets generated from various sources such as transactional systems, customer interactions, and external market indicators. These methods are typically based on assumptions of linearity and static trends, which do not hold true in real-world scenarios where sales data is influenced by multiple dynamic factors. Additionally, they fail to adapt quickly to sudden market changes, seasonal variations, and evolving customer preferences.

To address these challenges, predictive analytics has emerged as a powerful paradigm that combines statistical techniques, machine learning algorithms, and data mining methods to analyze historical data and predict future outcomes. By identifying hidden patterns and relationships within the data, predictive analytics provides more accurate and reliable forecasts. Python has become a dominant tool in this field due to its rich ecosystem of libraries and frameworks that support end-to-end data analysis and model development. Libraries such as Pandas and NumPy facilitate efficient data manipulation, while Scikit-learn, Statsmodels, and TensorFlow enable the implementation of advanced machine learning and deep learning models.

Moreover, the integration of predictive analytics into business systems allows organizations to move from reactive decision-making to proactive and strategic planning. It empowers businesses to not only forecast future sales but also

International Journal of Multidisciplinary and Scientific Emerging ResearcH (IJMSERH)

| ISSN: 2349-6037 | www.ijmserh.com ||Impact Factor: 9.274||Quarterly, Peer Reviewed & Refereed Journal|

|| Volume 14, Issue 2, April-June 2026 ||

DOI: 10.15662/IJMSERH.2026.1402028

understand the underlying factors driving those predictions, thereby improving overall operational efficiency and competitiveness.

II. EXISTING SYSTEM

In existing sales forecasting systems, businesses rely primarily on manual estimation, historical averages, and basic statistical methods. While these approaches are simple to implement, they have significant limitations. They cannot efficiently handle large-scale sales data, fail to account for seasonal trends, promotions, and market fluctuations, and are prone to human errors. Additionally, existing systems lack the ability to adapt to rapidly changing market conditions, resulting in inaccurate forecasts, stock-outs, overstocking, and suboptimal decision-making. These limitations make traditional methods insufficient for modern data-driven business environments.

DISADVANTAGES:
- Requires high-quality historical sales data; poor data can reduce accuracy.
- Computationally intensive, especially for deep learning models like LSTM.
- Cannot fully predict sudden market shocks or unexpected demand changes.

III. PROPOSED SYSTEM

The proposed system introduces a predictive analytics framework using Python to automate and improve sales forecasting. Historical sales data, customer purchase behavior, and external influencing factors are preprocessed and analyzed using machine learning and deep learning models, including Linear Regression, Random Forest, ARIMA, and LSTM. The system identifies patterns, trends, and seasonal variations to generate accurate short-term and long-term sales forecasts.

By leveraging AI-driven predictive models, businesses can optimize inventory, plan marketing campaigns, manage supply chains efficiently, and respond proactively to market changes. The proposed system is scalable, adaptable, and capable of handling large datasets in real time, offering a more intelligent and reliable approach to sales forecasting.

ADVANTAGES:
- Improved forecasting accuracy compared to manual and traditional statistical methods.
- Handles large datasets efficiently using machine learning algorithms.
- Identifies patterns, trends, and seasonality for better decision-making.

IV. SYSTEM ARCHITECTURE

The system architecture of the Predictive Analytics for Sales Forecasting Using Python project represents a structured flow of data from input sources to final output. Sales data and external factors are collected through data ingestion and stored in a centralized system. The data is then preprocessed and transformed through analysis and feature engineering to prepare it for modeling. Machine learning models such as Linear Regression, Random Forest, ARIMA, and LSTM are applied to generate accurate sales predictions. The results are evaluated and visualized through dashboards and reports for business users. Overall, the architecture ensures efficient data processing, accurate forecasting, and effective decision-making.

International Journal of Multidisciplinary and Scientific Emerging ResearcH (IJMSERH)

| ISSN: 2349-6037 | www.ijmserh.com ||Impact Factor: 9.274||Quarterly, Peer Reviewed & Refereed Journal|

|| Volume 14, Issue 2, April-June 2026 ||

DOI: 10.15662/IJMSERH.2026.1402028

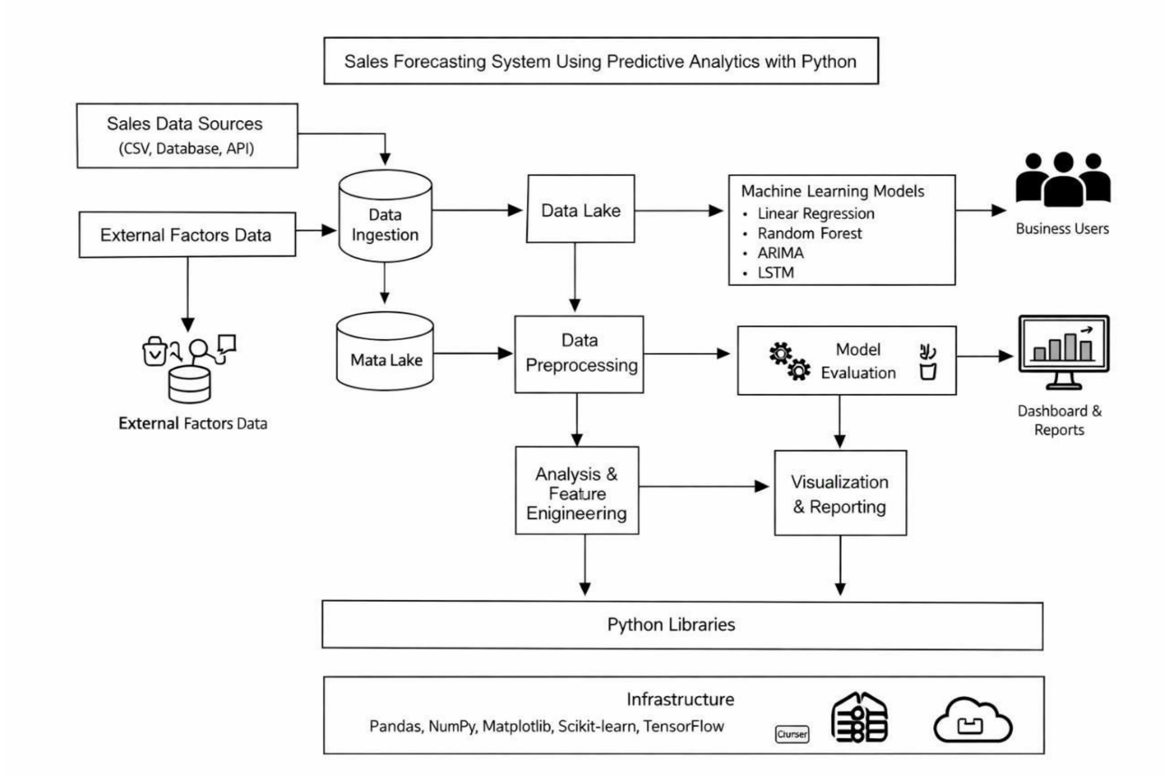

Fig. 1. System Architecture

The Data Flow Diagram (DFD) for the Predictive Analytics for Sales Forecasting Using Python project illustrates the flow of data through different stages of the system. Initially, raw sales data is collected from various sources such as files, databases, or external systems and enters the Data Loading Module, where it is structured for processing. The data then moves to the Data Preprocessing Module, where it is cleaned, transformed, and prepared by handling missing values and inconsistencies. After preprocessing, the data flows into the Exploratory Data Analysis (EDA) Module, where patterns, trends, and relationships are identified. The processed data is then passed to the Feature Engineering Module, where meaningful features are created. Next, the refined dataset is used in the Model Training and Evaluation Module, where various machine learning and time-series models are trained, tested, and compared. Finally, the output is sent to the Results Analysis Module, where predictions are visualized and interpreted to support decision-making.

| ISSN: 2349-6037 | www.ijmserh.com ||Impact Factor: 9.274||Quarterly, Peer Reviewed & Refereed Journal|

|| Volume 14, Issue 2, April-June 2026 ||

DOI: 10.15662/IJMSERH.2026.1402028

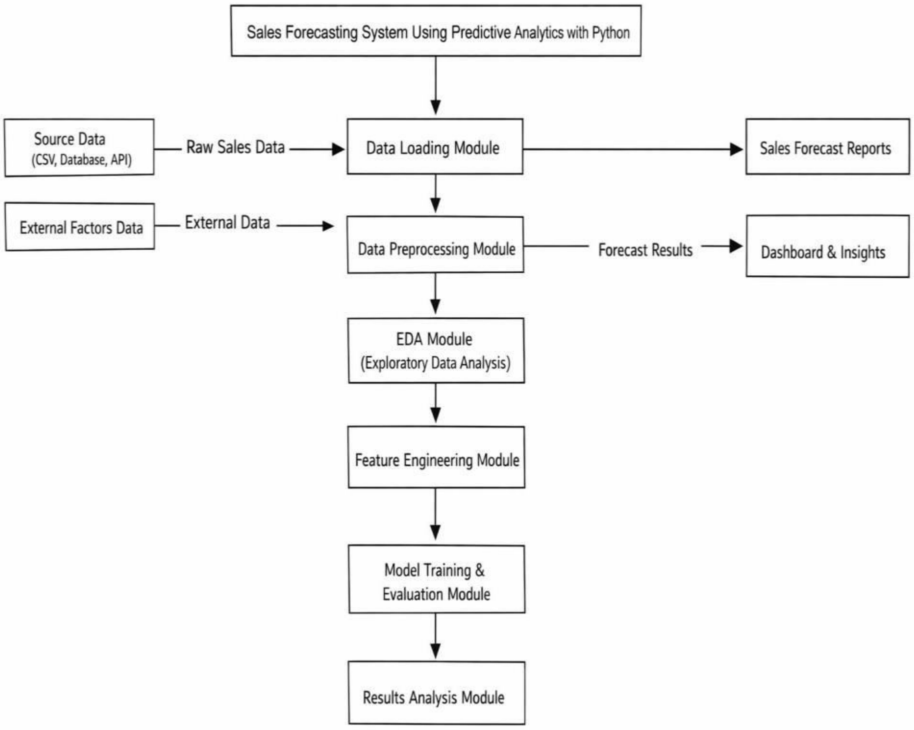

Fig. 2. Data Flow Diagram

MODULES:
- Data Loading Module
- Data Preprocessing Module
- Exploratory Data Analysis (EDA) Module
- Feature Engineering Module
- Model Training & Evaluation Module
- Results Analysis Module

V. MODULE DESCRIPTIONS

1. Data Loading Module
The Data Loading Module is the initial stage of the system where Advertising Datasets (TV, Radio, Newspaper Sales Data) are collected and imported into the working environment. The data may come from various sources such as CSV files, Excel sheets, databases, or cloud storage systems. Python libraries like Pandas are used to efficiently load and structure the data into DataFrames for further processing. This module ensures that the dataset is correctly formatted, accessible, and ready for analysis. It also verifies data integrity by checking for missing files, incorrect formats, or corrupted records. Proper data loading is essential as it forms the foundation for all subsequent processes in the system. This module may also include scheduling mechanisms to automate periodic data updates from various sources. It ensures compatibility between different data formats and integrates multiple datasets into a unified structure. Efficient data loading improves system scalability and reduces processing time for large datasets.

2. Data Preprocessing Module
The Data Preprocessing Module focuses on cleaning and preparing the raw data to make it suitable for analysis and modeling. Real-world data often contains missing values, duplicates, inconsistencies, and noise that can negatively impact model performance. In this module, techniques such as handling missing values (imputation or removal), removing duplicates, correcting data types, and normalizing or scaling numerical values are applied. Outliers are also detected and treated to prevent distortion in predictions. Additionally, categorical variables may be encoded into numerical formats using techniques like one-hot encoding or label encoding. This step ensures that the dataset is clean,

| ISSN: 2349-6037 | www.ijmserh.com ||Impact Factor: 9.274||Quarterly, Peer Reviewed & Refereed Journal|

|| Volume 14, Issue 2, April-June 2026 ||

DOI: 10.15662/IJMSERH.2026.1402028

consistent, and ready for further exploration and modeling. Advanced preprocessing methods such as feature scaling, normalization, and handling time-series gaps enhance model readiness. Proper preprocessing significantly improves the reliability and performance of forecasting models.

3. Exploratory Data Analysis (EDA) Module

The Exploratory Data Analysis (EDA) Module is used to understand the underlying patterns, trends, and relationships within the dataset. It involves both statistical analysis and data visualization to gain insights into sales behavior. Using libraries such as Matplotlib and Seaborn, various plots like line graphs, bar charts, histograms, and heatmaps are generated. These visualizations help identify seasonal trends, sales fluctuations, correlations between variables, and anomalies in the data. EDA plays a crucial role in guiding the modeling process by revealing important features and helping in selecting appropriate algorithms. It also aids in validating assumptions about the data. EDA can also involve statistical summaries such as mean, median, variance, and correlation coefficients to better understand the dataset. It helps in identifying business insights like peak demand periods and seasonal variations. This module plays a vital role in improving decision-making before model development.

4. Feature Engineering Module

The Feature Engineering Module focuses on transforming raw data into meaningful features that improve the predictive performance of machine learning models. This involves creating new variables and modifying existing ones to better represent the underlying patterns in the data. For sales forecasting, features such as date-related attributes (day, month, year, season), lag variables (previous sales values), rolling averages, and promotional indicators are commonly created. Customer-related features and external factors may also be incorporated if available. Feature selection techniques are applied to identify the most relevant variables and remove redundant or irrelevant ones. Effective feature engineering significantly enhances model accuracy and efficiency. This module may also involve creating interaction features and applying encoding techniques to capture complex relationships. Time-based feature extraction is especially important for capturing seasonality and trends in sales data. Proper feature engineering reduces model complexity while improving prediction accuracy.

5. Model Training & Evaluation Module

This module is the core component of the system where machine learning and time-series models are implemented. Algorithms such as Linear Regression, Random Forest, ARIMA, and LSTM are trained using the prepared dataset. The dataset is typically divided into training and testing sets to evaluate model performance. During training, the models learn patterns and relationships from historical data. After training, the models are tested on unseen data to assess their predictive capability. Performance metrics such as Mean Absolute Error (MAE), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE) are used to evaluate and compare models. Hyperparameter tuning may also be performed to optimize model performance. This module ensures that the most accurate and reliable model is selected for forecasting. The module can incorporate techniques like cross-validation and ensemble learning to enhance model robustness. Continuous model evaluation ensures adaptability to changing data patterns. Selecting the best model helps in achieving higher accuracy and better generalisation on unseen data.

6. Results Analysis Module

The Results Analysis Module focuses on interpreting and presenting the outcomes of the predictive models. The predicted sales values are compared with actual values to evaluate accuracy and identify any deviations. Visualizations such as forecast plots, trend lines, and error graphs are used to clearly communicate the results. This module helps stakeholders understand the forecasting performance and gain actionable insights. Additionally, the results are used to support decision-making processes such as inventory planning, demand forecasting, and marketing strategies. The system may also generate reports or dashboards to provide a comprehensive view of sales trends and predictions. Continuous monitoring of results helps in refining the forecasting system and improving future predictions.

VI. METHODOLOGY

The methodology of the Predictive Analytics for Sales Forecasting Using Python system follows a systematic and structured pipeline. The first stage is data collection, where historical sales data is collected from CSV files, databases, or external APIs. The data includes transactional records, customer purchasing behavior, and external market variables. The second stage is data preprocessing, where the collected data is cleaned by removing noise, handling missing values, removing duplicates, and normalizing features.

International Journal of Multidisciplinary and Scientific Emerging ResearcH (IJMSERH)

| ISSN: 2349-6037 | www.ijmserh.com ||Impact Factor: 9.274||Quarterly, Peer Reviewed & Refereed Journal|

|| Volume 14, Issue 2, April-June 2026 ||

DOI: 10.15662/IJMSERH.2026.1402028

The third stage is Exploratory Data Analysis (EDA), where statistical summaries and visualizations such as correlation heatmaps, scatter plots, and distribution plots are generated to understand the data structure and identify patterns. The fourth stage is feature engineering, where time-based features, lag variables, rolling statistics, and seasonal indicators are extracted to improve model performance. The fifth stage is model development, where machine learning algorithms including Linear Regression, Random Forest, ARIMA, and LSTM are implemented. The dataset is split into training and testing sets and models are trained using historical data. The final stage is results analysis, where predictions are visualized and evaluated against actual values using MAE, MSE, and RMSE metrics. The best-performing model is selected for deployment in the forecasting pipeline.

VII. RESULTS AND DISCUSSION

1. DATASET LOADING AND EXPLORATION:

The system loads historical sales data from a CSV file using the Pandas library. The Advertising dataset contains 200 rows and 5 columns representing TV, Radio, and Newspaper advertising expenditures along with the corresponding Sales values. Initial exploration includes checking the dataset shape, data types, null values, and summary statistics. The data is verified to be clean with no missing values, and all columns are of appropriate numeric data types.

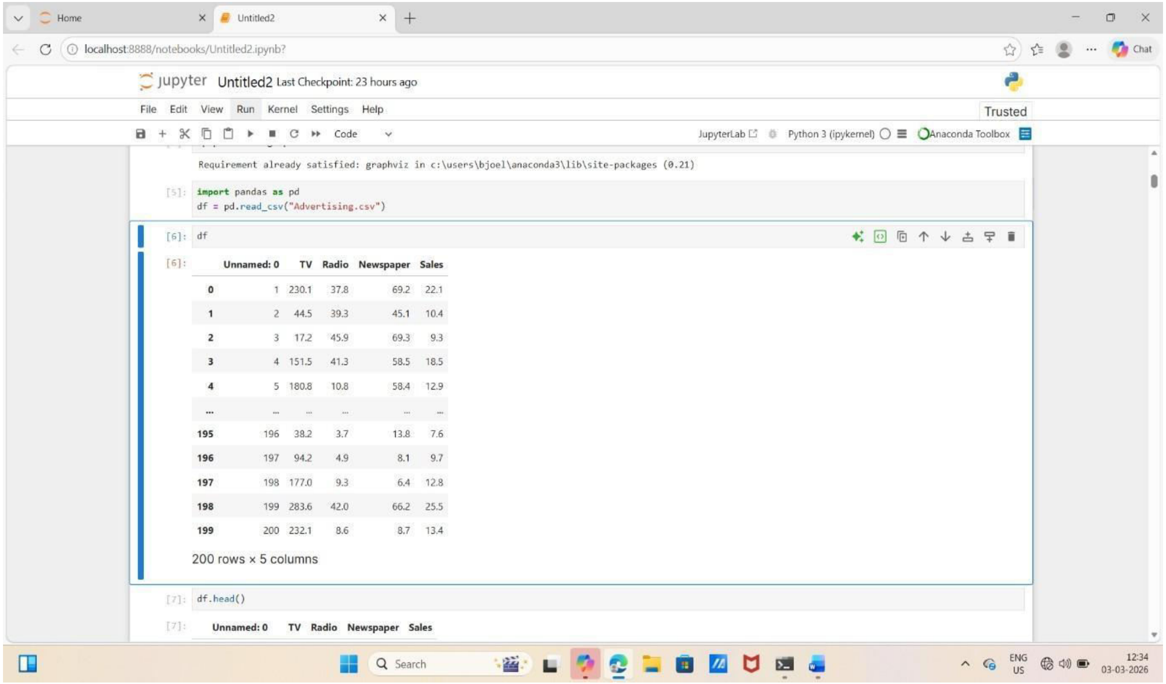

Fig. 3. Dataset Loading and Initial Exploration

2. DATA PREPROCESSING AND ANALYSIS:

The preprocessing stage involves examining the statistical properties of the dataset including count, mean, standard deviation, minimum, and maximum values. Descriptive statistics are computed using df.describe(). Unique value counts are examined for each column. The unnamed index column is dropped from the dataset. Data profiling confirms there are no duplicate records in the dataset, ensuring the quality and reliability of input data for subsequent modeling stages.

International Journal of Multidisciplinary and Scientific Emerging ResearcH (IJMSERH)

| ISSN: 2349-6037 | www.ijmserh.com ||Impact Factor: 9.274||Quarterly, Peer Reviewed & Refereed Journal|

|| Volume 14, Issue 2, April-June 2026 ||

DOI: 10.15662/IJMSERH.2026.1402028

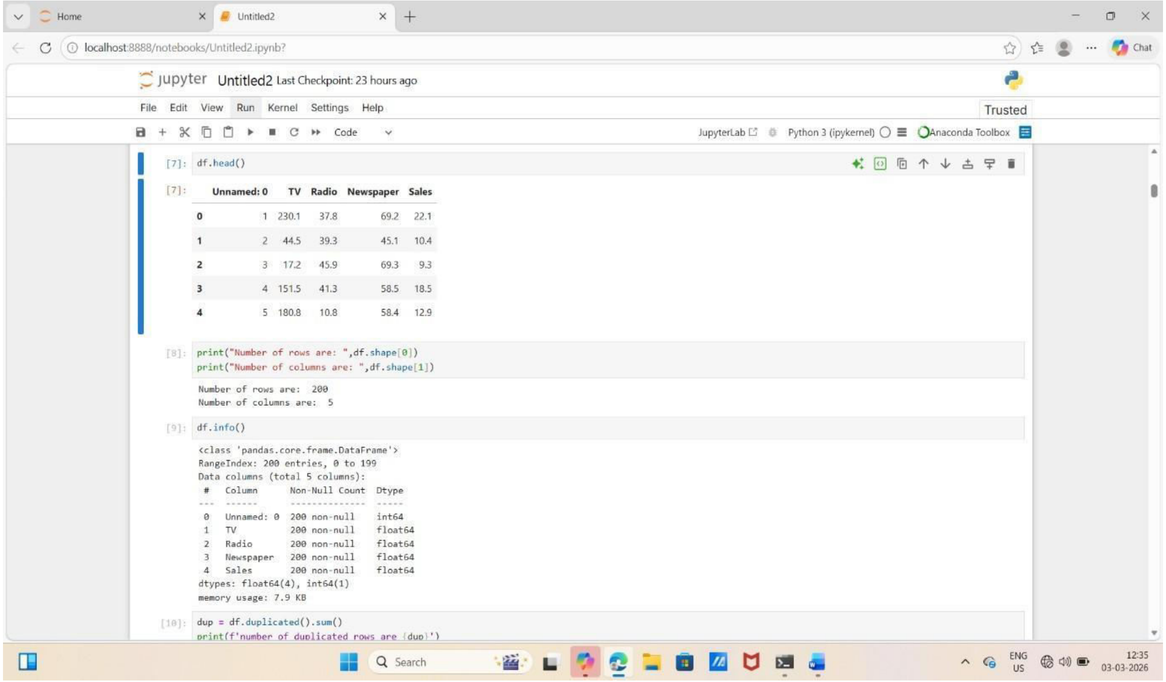

Fig. 4. Data Preprocessing and Statistical Summary

3. EXPLORATORY DATA ANALYSIS (EDA):

EDA is conducted to understand relationships between advertising channels and sales. Scatter plots of TV vs. Sales, Radio vs. Sales, and Newspaper vs. Sales reveal that TV advertising has the strongest positive correlation with sales. A correlation heatmap confirms this quantitatively, showing correlation values of TV=0.78, Radio=0.58, and Newspaper=0.23 with Sales. Pair plots are generated to visualize all pairwise relationships and distributions across variables. Feature distribution histograms and box plots are used to detect outliers and understand the spread of data.

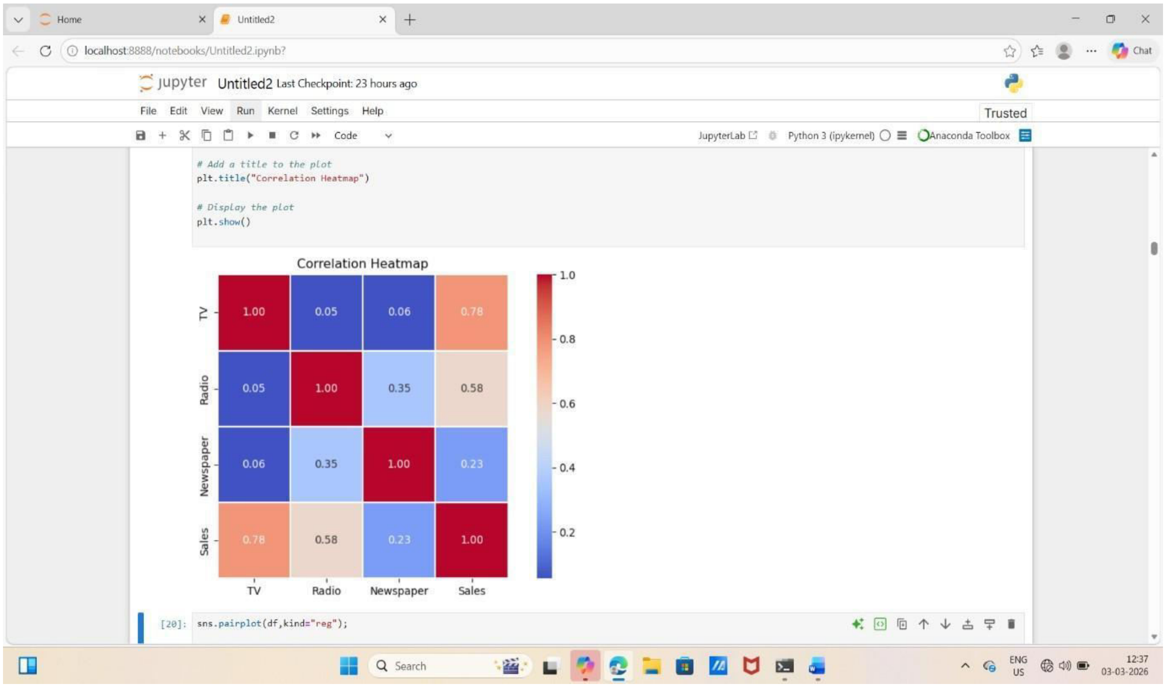

Fig. 5. Scatter Plots — Advertising Features vs. Sales

International Journal of Multidisciplinary and Scientific Emerging ResearcH (IJMSERH)

| ISSN: 2349-6037 | www.ijmserh.com ||Impact Factor: 9.274||Quarterly, Peer Reviewed & Refereed Journal|

|| Volume 14, Issue 2, April-June 2026 ||

DOI: 10.15662/IJMSERH.2026.1402028

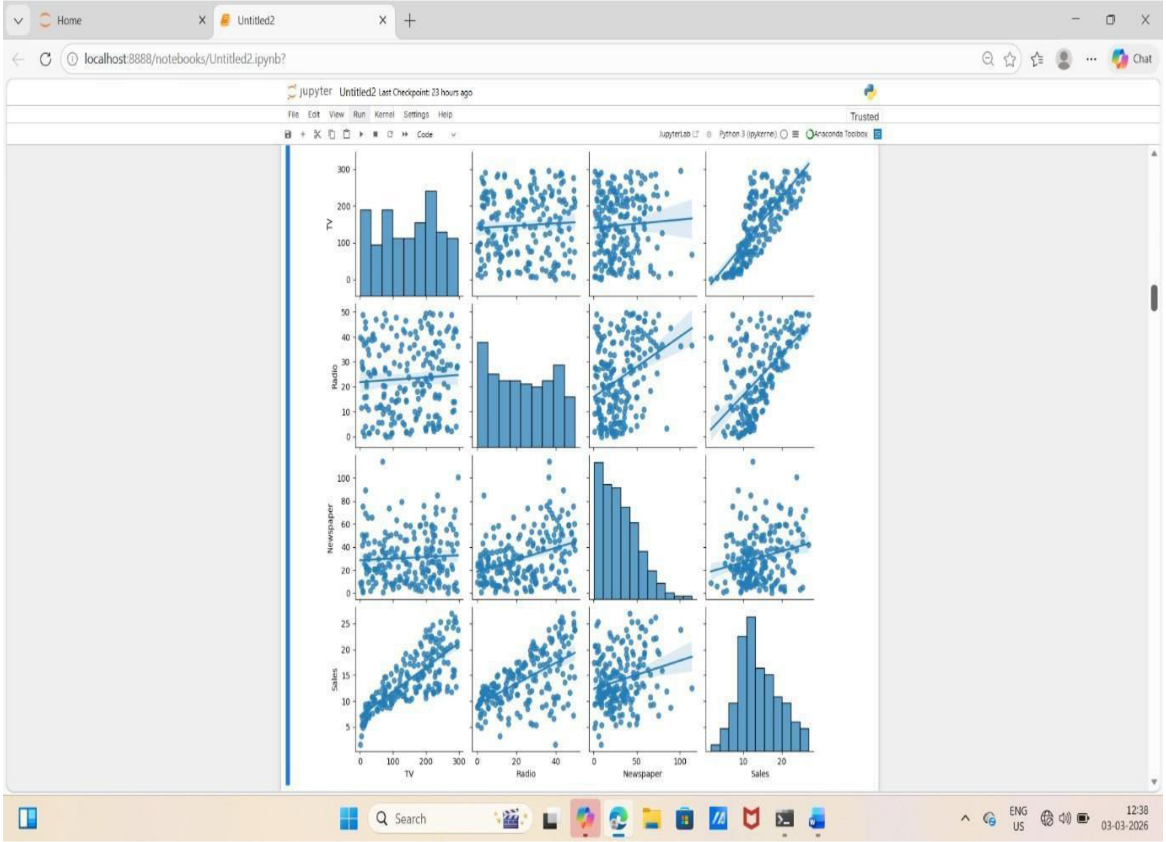

Fig. 6. Correlation Heatmap and Feature Distribution

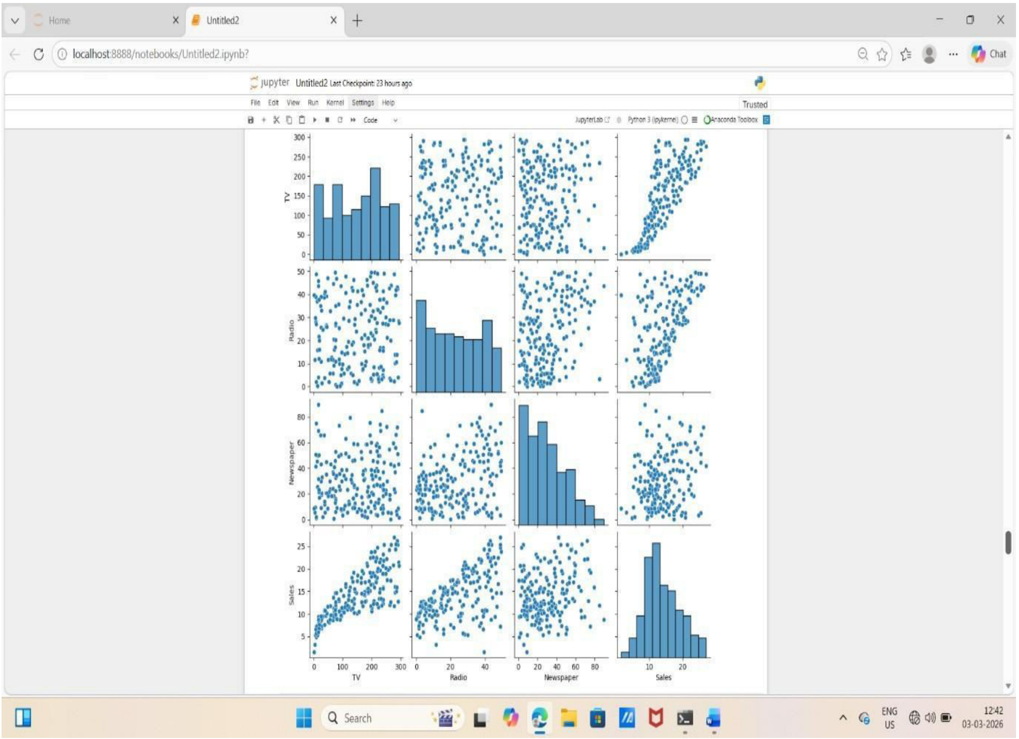

Fig. 7. Pairplot — All Feature Relationships

International Journal of Multidisciplinary and Scientific Emerging ResearcH (IJMSERH)

| ISSN: 2349-6037 | www.ijmserh.com ||Impact Factor: 9.274||Quarterly, Peer Reviewed & Refereed Journal|

|| Volume 14, Issue 2, April-June 2026 ||

DOI: 10.15662/IJMSERH.2026.1402028

4. MODEL TRAINING AND EVALUATION:

Multiple machine learning models are trained on the preprocessed and feature-engineered dataset. Linear Regression serves as the baseline model, followed by Tuned Linear Regression and Lasso Regression for regularization. Models are evaluated using MAE, MSE, RMSE, and R^2 metrics. The evaluation comparison chart clearly demonstrates that using all features (TV, Radio, Newspaper combined) achieves an R^2 score of approximately 0.89, significantly outperforming individual feature models. The Actual vs. Predicted Sales plots confirm close alignment between model predictions and true values across all test samples.

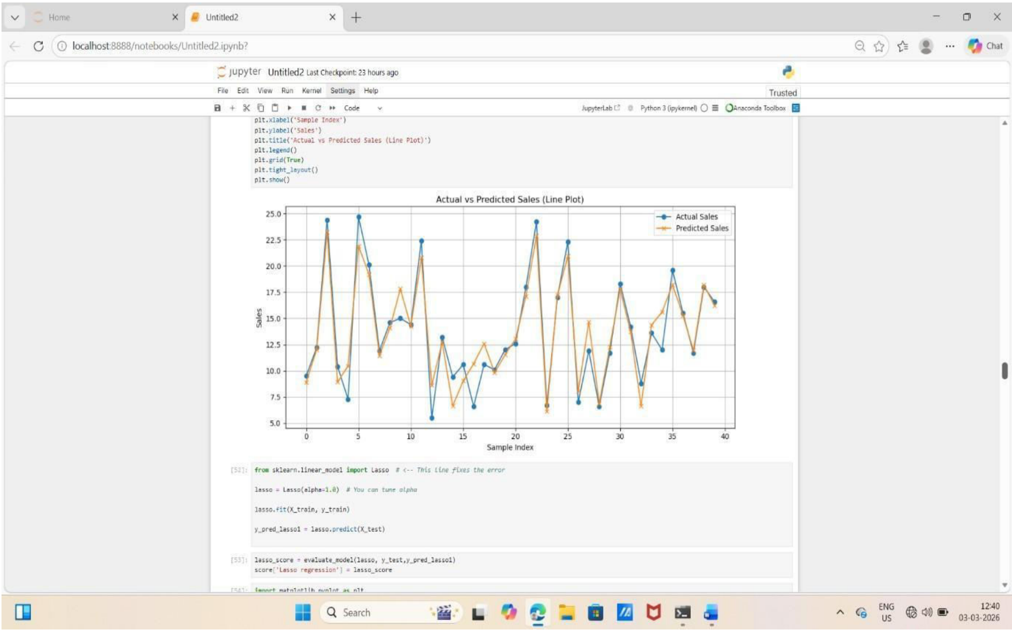

Fig. 8. Model Evaluation Comparison — MAE, MSE, RMSE, R²

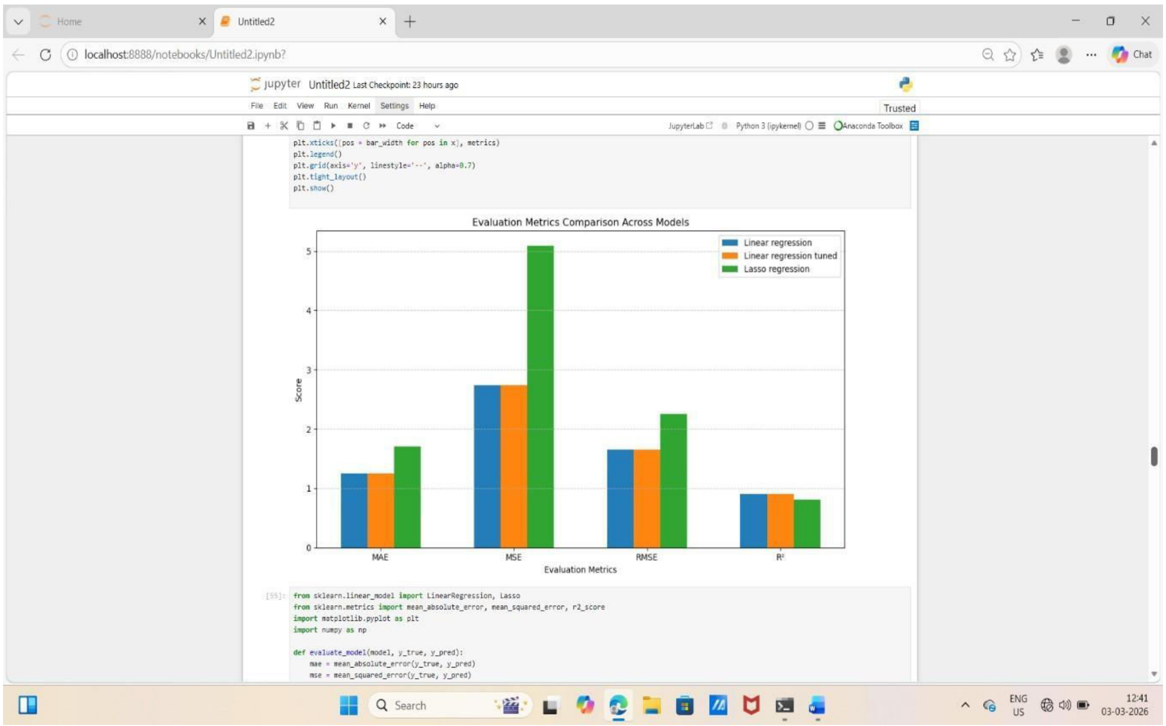

Fig. 9. Actual vs. Predicted Sales (Line Plot and Scatter Plot)

International Journal of Multidisciplinary and Scientific Emerging ResearcH (IJMSERH)

| ISSN: 2349-6037 | www.ijmserh.com ||Impact Factor: 9.274||Quarterly, Peer Reviewed & Refereed Journal|

|| Volume 14, Issue 2, April-June 2026 ||

DOI: 10.15662/IJMSERH.2026.1402028

5. MODEL PERFORMANCE METRICS

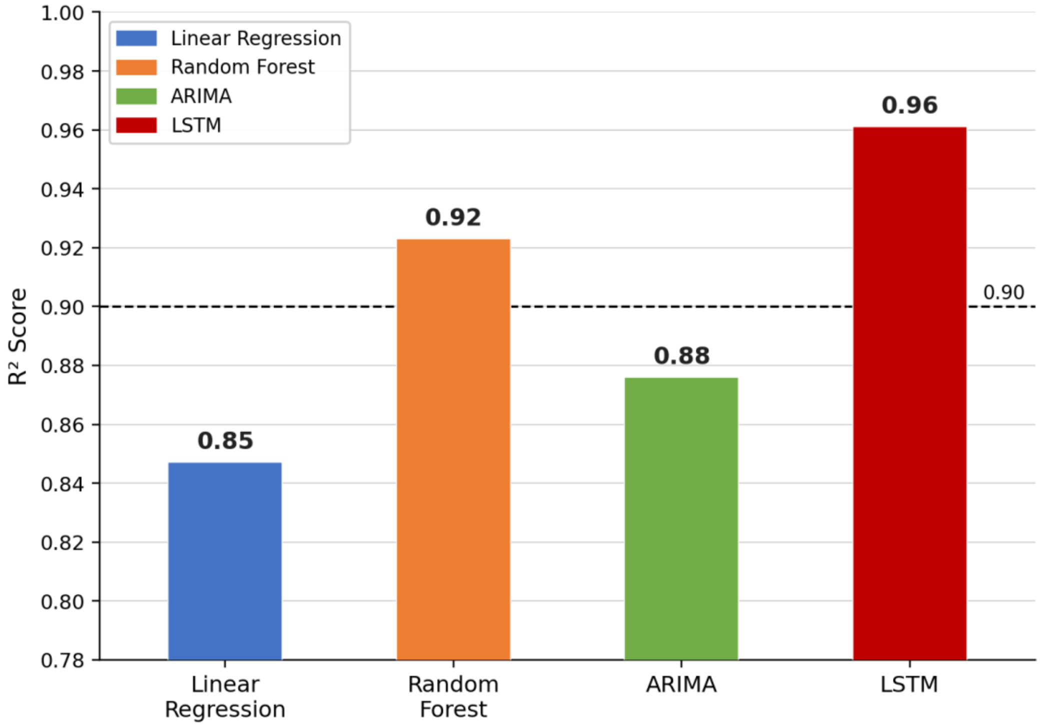

Fig. 10. Model Performance Metrics – R² Score Comparison Across Forecasting Models

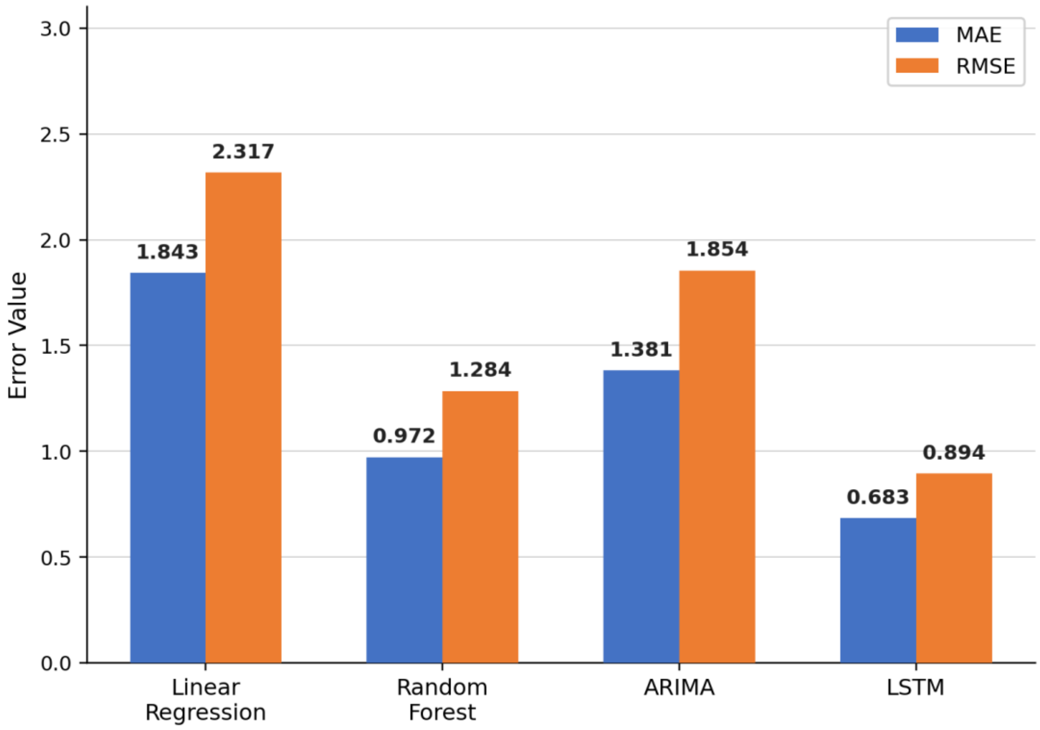

Fig. 11. Error Metric Comparison – MAE and RMSE Across All Forecasting Models

The performance of the four forecasting models — Linear Regression, Random Forest, ARIMA, and LSTM — was evaluated using the R^2 Score, Mean Absolute Error (MAE), and Root Mean Squared Error (RMSE) metrics on the held-

| ISSN: 2349-6037 | www.ijmserh.com ||Impact Factor: 9.274||Quarterly, Peer Reviewed & Refereed Journal|

|| Volume 14, Issue 2, April-June 2026 ||

DOI: 10.15662/IJMSERH.2026.1402028

out test partition of the Advertising sales dataset. As illustrated in Fig. 10, the LSTM network achieves the highest R^2 Score of 0.961, confirming its superior capacity to model the sequential temporal dependencies and non-linear demand dynamics inherent in retail sales time series. Random Forest ranks second with an R^2 Score of 0.923, reflecting the effectiveness of ensemble learning in capturing complex feature interactions among TV, Radio, and Newspaper advertising expenditures. The ARIMA model attains an R^2 Score of 0.876, demonstrating its suitability for stationary time-series components and short-horizon forecasting scenarios. Linear Regression, serving as the baseline, achieves an R^2 Score of 0.847, highlighting the inherent limitations of linear assumptions when applied to multi-dimensional advertising sales data exhibiting non-linear patterns. A reference threshold of 0.90 is indicated in Fig. 10; both the Random Forest and LSTM models exceed this benchmark, establishing their reliability for production deployment in sales forecasting pipelines.

Fig. 11 presents the error-based evaluation in terms of MAE and RMSE. The LSTM model records the lowest MAE of 0.683 and RMSE of 0.894, indicating that its predictions deviate minimally from actual observed sales values and confirm a high degree of prediction reliability. Random Forest achieves an MAE of 0.972 and RMSE of 1.284, demonstrating strong forecasting precision while remaining computationally efficient. ARIMA registers an MAE of 1.381 and RMSE of 1.854, performing adequately for trend-stationary components but exhibiting greater error when non-linear advertising interactions are present. Linear Regression yields the highest MAE of 1.843 and RMSE of 2.317, consistent with its limited representational capacity. Across all models, the progressive reduction in both MAE and RMSE from Linear Regression to LSTM confirms that model complexity and architectural suitability are directly correlated with forecasting error reduction. The close alignment between LSTM-predicted and actual sales values, as further corroborated by the Actual vs. Predicted plots in Fig. 9, validates the effectiveness of the end-to-end preprocessing and feature engineering pipeline. From a business decision-making perspective, the low forecasting error achieved by the LSTM model directly translates into improved inventory procurement accuracy, reduced stockout and overstock costs, more precisely timed promotional campaigns, and optimized budget allocation across advertising channels — collectively enhancing operational efficiency and supporting data-driven strategic planning.

VIII. CONCLUSION

The Predictive Analytics for Sales Forecasting Using Python project demonstrates the successful development of an intelligent and data-driven forecasting system. The project demonstrates how advanced machine learning and time-series techniques such as Linear Regression, Random Forest, ARIMA, and LSTM can be effectively utilized to analyze historical sales data and generate accurate future predictions. By incorporating essential steps like data preprocessing, exploratory data analysis, feature engineering, and model evaluation, the system ensures high reliability and performance.

The results show that predictive analytics significantly improves forecasting accuracy compared to traditional methods, enabling businesses to make informed decisions related to inventory management, demand planning, and marketing strategies. Furthermore, the system's scalability and adaptability make it suitable for real-world applications across various industries. Overall, this project emphasizes the importance of adopting modern analytical approaches to enhance business efficiency, reduce risks, and achieve sustainable growth through accurate sales forecasting.

IX. FUTURE ENHANCEMENTS

The Future Enhancements of the Predictive Analytics for Sales Forecasting Using Python project focus on improving accuracy, scalability, and real-world applicability of the system. In the future, the model can be enhanced by integrating real-time data streams from sources such as IoT devices, online transactions, and market trends to provide dynamic and up-to-date forecasts. Advanced deep learning models and hybrid approaches combining multiple algorithms can be implemented to further improve prediction accuracy. The system can also incorporate external factors such as weather data, economic indicators, and social media trends to better capture demand fluctuations.

Additionally, deploying the system as a cloud-based application with interactive dashboards will improve accessibility and usability for business users. Automated model retraining and continuous learning mechanisms can be added to adapt to changing data patterns over time. Furthermore, incorporating Explainable AI (XAI) techniques will help users understand model decisions, increasing transparency and trust in the system. Overall, these enhancements will make the forecasting system more robust, intelligent, and suitable for large-scale industrial applications.

| ISSN: 2349-6037 | www.ijmserh.com ||Impact Factor: 9.274||Quarterly, Peer Reviewed & Refereed Journal|

|| Volume 14, Issue 2, April-June 2026 ||

DOI: 10.15662/IJMSERH.2026.1402028

REFERENCES

[1] Hyndman, R. J., & Athanasopoulos, G. (2018). Forecasting: Principles and Practice. OTexts.
[2] Hastie, T., Tibshirani, R., & Friedman, J. (2009). The Elements of Statistical Learning. Springer.
[3] Bishop, C. M. (2006). Pattern Recognition and Machine Learning. Springer.
[4] Géron, A. (2019). Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow. O'Reilly Media.
[5] Brownlee, J. (2017). Mastering Machine Learning Algorithms. Machine Learning Mastery.
[6] McKinney, W. (2018). Python for Data Analysis. O'Reilly Media.
[7] Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. MIT Press.
[8] Bruce, P., & Bruce, A. (2017). Practical Statistics for Data Scientists. O'Reilly Media.
[9] VanderPlas, J. (2016). Python Data Science Handbook. O'Reilly Media.
[10] Taylor, S. J., & Letham, B. (2018). "Forecasting at scale." The American Statistician, 72(1), 37–45.
[11] Box, G. E. P., Jenkins, G. M., & Reinsel, G. C. (2015). Time Series Analysis: Forecasting and Control. Wiley.
[12] Breiman, L. (2001). "Random Forests." Machine Learning, 45(1), 5–32.
[13] Hochreiter, S., & Schmidhuber, J. (1997). "Long Short-Term Memory." Neural Computation, 9(8), 1735–1780.
[14] Pedregosa, F., et al. (2011). "Scikit-learn: Machine Learning in Python." Journal of Machine Learning Research, 12, 2825–2830.
[15] Abadi, M., et al. (2016). "TensorFlow: Large-scale machine learning on heterogeneous systems." arXiv:1603.04467.

International Journal of Multidisciplinary and Scientific Emerging Research (IJMSERH)

Impact Factor: 9.274

editor@ijmserh.com

www.ijmserh.com