Sales Forecasting and Demand Prediction Project using Machine Learning

Wael Ahmed Mohammed Rabie Al_naqiti
Faculty of computers and Data Science
waeeel989@gmail.com

Muhammad Khalid Muhammad Ali
Faculty of Computer & Information
m.b.s.012106.0@gmail.com

Ahmed Essam Mostafa
Faculty on Al RYADA UNIVERSITY
ahesmoal@gmail.com

Norhan Mostafa Hassan Ali
Faculty of computers & information
norhanmostaga456@gmail.com

Anas Mohammed Ebrahim Desoky
Faculty of Ai and Computer Science
anasdesoky0@gmail.com

Ahmed Mosaad Yehia
Faculty of Computers & Ai
ahmedmossaad00@gmail.com

***Abstract*—**The **Sales Forecasting and Demand Prediction Project** aims to develop a robust machine learning system capable of predicting future sales and product demand using historical data and external influencing factors. By leveraging time series analysis, feature engineering, and advanced forecasting models, the project provides accurate and data-driven predictions that help businesses optimize inventory management, reduce stockouts, and improve marketing and operational strategies. The end-to-end pipeline includes data exploration, preprocessing, model development, MLOps integration, deployment, and continuous monitoring to ensure scalability, reliability, and long-term performance.

I. Introduction

Forecasting future sales and product demand is a critical component of effective business planning. Accurate predictions enable organizations to manage inventory efficiently, avoid overstocking or understocking, allocate resources more effectively, and enhance customer satisfaction. With the increasing availability of structured and unstructured data, machine learning has become a powerful tool for extracting meaningful patterns and generating reliable forecasts.

The end-to-end pipeline includes data exploration, preprocessing, model development, MLOps integration, deployment, and continuous monitoring to ensure scalability, reliability, and long-term performance.

This project focuses on building a comprehensive sales forecasting and demand prediction system through a structured, milestone-based approach. Beginning with data collection and exploratory analysis, the project progresses through advanced feature engineering, model selection, training, evaluation, and deployment using modern MLOps practices. The model leverages both internal historical data and external variables—such as promotions, weather conditions, holidays, and economic indicators—to improve forecast accuracy.

By the end of the project, the outcome is a fully functional forecasting model deployed as an API or web application, supported by monitoring tools and documentation to ensure continued performance and business value. This system empowers decision-makers with insights that drive smarter marketing campaigns, optimized staffing, supply chain efficiency, and improved strategic planning.

.

II. Methodology

The methodology of the Sales Forecasting and Demand Prediction Project follows a structured, milestone-driven workflow designed to ensure accurate, scalable, and actionable forecasting results. The process integrates data science, machine learning, and MLOps practices to build an end-to-end forecasting system. The methodology includes the following phases:

A. Data Collection

The project begins by gathering historical sales and demand data from reliable sources which is Kaggle . The dataset includes essential features such as product information, customer behavior, promotional activities, seasonality indicators, holidays, weather data, and other economic factors. Collecting diverse and rich data ensures that the forecasting model captures all key variables that influence demand.

B. Data Exploration and Preprocessing

After data acquisition, exploratory data analysis (EDA) is performed to understand the structure and quality of the dataset. This involves:

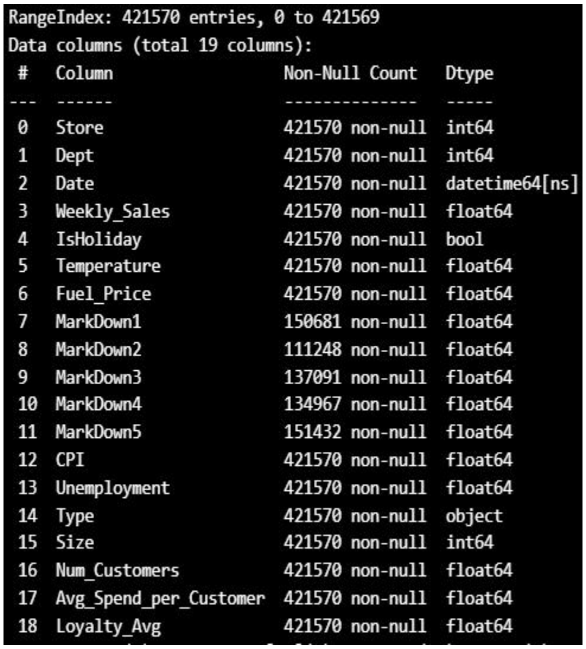
1) Encoding Categorical Data: The dataset contains several categorical variables such as product categories, store types, promotion flags, and seasonal labels. To convert these into a numerical format interpretable by machine learning algorithms, encoding techniques were applied.

Binary features (e.g., Promotion: Yes/No) were transformed using **Label Encoding**.
Multi-class categorical features (such as **Fuel Category** and the column **Type**) were processed using **One-Hot Encoding** to avoid introducing artificial order.

This encoding ensures that the forecasting model can correctly interpret categorical information without introducing unintended bias.

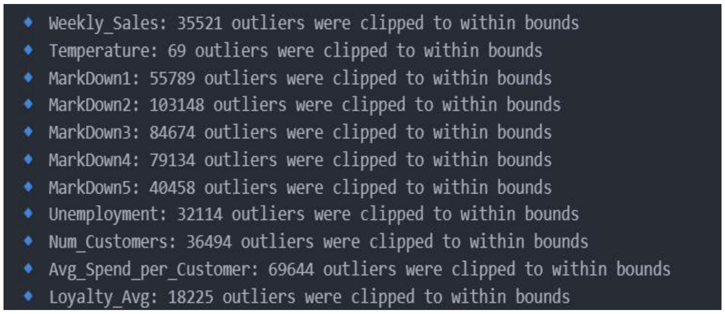
2) *Splitting the Dataset:* The next stage in machine learning data preprocessing is to split the dataset. A machine learning model's dataset should be split into two parts: training and testing.

We divided the data into an 80:20 split. This means that we use 80% of the data to train the model while keeping the remaining 20% for testing. We take all the 20 independent



```
RangeIndex: 421570 entries, 0 to 421569
Data columns (total 19 columns):
 #   Column                 Non-Null Count   Dtype
---  ------                 --------------   -----
 0   Store                  421570 non-null  int64
 1   Dept                   421570 non-null  int64
 2   Date                   421570 non-null  datetime64[ns]
 3   Weekly_Sales           421570 non-null  float64
 4   IsHoliday              421570 non-null  bool
 5   Temperature            421570 non-null  float64
 6   Fuel_Price             421570 non-null  float64
 7   MarkDown1              150681 non-null  float64
 8   MarkDown2              111248 non-null  float64
 9   MarkDown3              137091 non-null  float64
 10  MarkDown4              134967 non-null  float64
 11  MarkDown5              151432 non-null  float64
 12  CPI                    421570 non-null  float64
 13  Unemployment           421570 non-null  float64
 14  Type                   421570 non-null  object
 15  Size                   421570 non-null  int64
 16  Num_Customers          421570 non-null  float64
 17  Avg_Spend_per_Customer 421570 non-null  float64
 18  Loyalty_Avg            421570 non-null  float64
```

Fig. 1. Show Range index,data coloums and dtype.



```
◆ Weekly_Sales: 35521 outliers were clipped to within bounds
◆ Temperature: 69 outliers were clipped to within bounds
◆ MarkDown1: 55789 outliers were clipped to within bounds
◆ MarkDown2: 103148 outliers were clipped to within bounds
◆ MarkDown3: 84674 outliers were clipped to within bounds
◆ MarkDown4: 79134 outliers were clipped to within bounds
◆ MarkDown5: 40458 outliers were clipped to within bounds
◆ Unemployment: 32114 outliers were clipped to within bounds
◆ Num_Customers: 36494 outliers were clipped to within bounds
◆ Avg_Spend_per_Customer: 69644 outliers were clipped to within bounds
◆ Loyalty_Avg: 18225 outliers were clipped to within bounds
```

Fig. 2. Outliers Clipping

C. Exploratory Data Analysis:

Exploratory Data Analysis was conducted to understand the underlying structure of the sales dataset and summarize its key characteristics. Using statistical summaries and visualizations such as line plots, heatmaps, bar charts, and seasonal decomposition, EDA provided valuable insights into trends, seasonality, and demand fluctuations over time.
EDA helps data scientists by:
Increasing the understanding of sales patterns and product behavior
Detecting seasonal and cyclic trends
Identifying correlations between promotions, holidays, and sales volume
Gaining a clearer understanding of the forecasting problem problem

D. Hyperparameter tuning by Random Search CV

To improve the performance of forecasting models, hyperparameter tuning was performed using RandomSearchcCV.The objective was to determine the optimal set of parameters that achieve the highest forecasting accuracy while minimizing error metrics such as MAE, and RMSE.

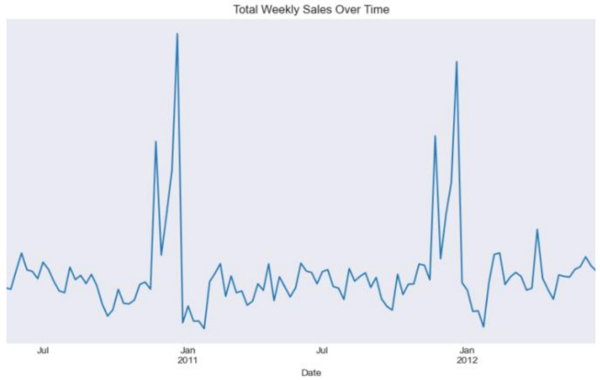


Fig. 3. Total weekly sales overtime

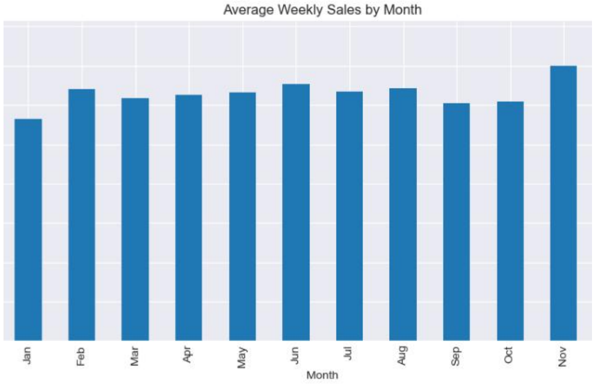


Fig. 4. Average weekly sales month

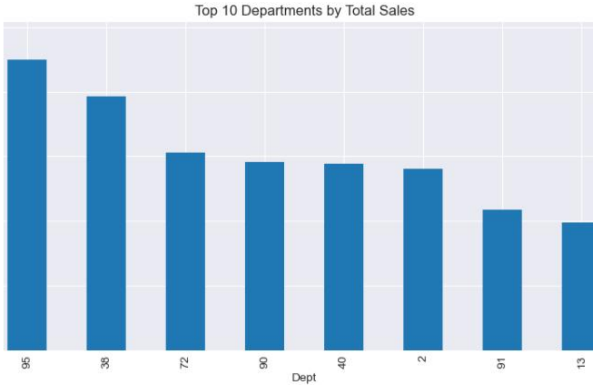


Fig. 5. Top10 Department by total sales

III. Implementation

With advancements in machine learning and computational power, predictive modeling for sales forecasting has become more efficient and scalable. Random Search CV was applied to identify the best-performing configuration for each model.

The models implemented include:

A. Linear Regression

Linear Regression is a fundamental statistical and machine-learning technique used for predicting continuous outcomes by modeling the linear relationship between input features and the target variable.

It works by fitting a best-fit line that minimizes the error between predicted and actual values.

B. **Random Forest Regressor**

Random Forest is an ensemble learning method that generates predictions by averaging outputs from multiple decision trees.
It is effective in capturing complex interactions between features such as promotions, product attributes, and economic indicators.
Random Forest helps reduce overfitting and provides insights into feature importance, highlighting which factors influence sales the most.

C. **Gradient Boosting Models (XGBoost / LightGBM)**

Boosting algorithms iteratively improve model performance by focusing on errors from previous iterations.
These models can capture non-linear relationships and are known for their high accuracy in tabular forecasting tasks.
They perform particularly well when incorporating engineered features such as lag values, rolling averages, and seasonal indicators.

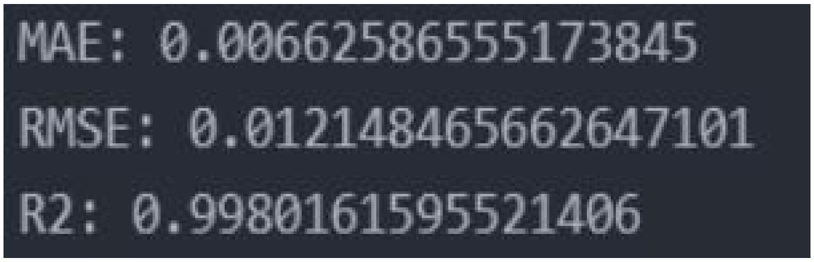
IV.Result and Discussion

To assess the performance of the forecasting models, several evaluation metrics were used, including:

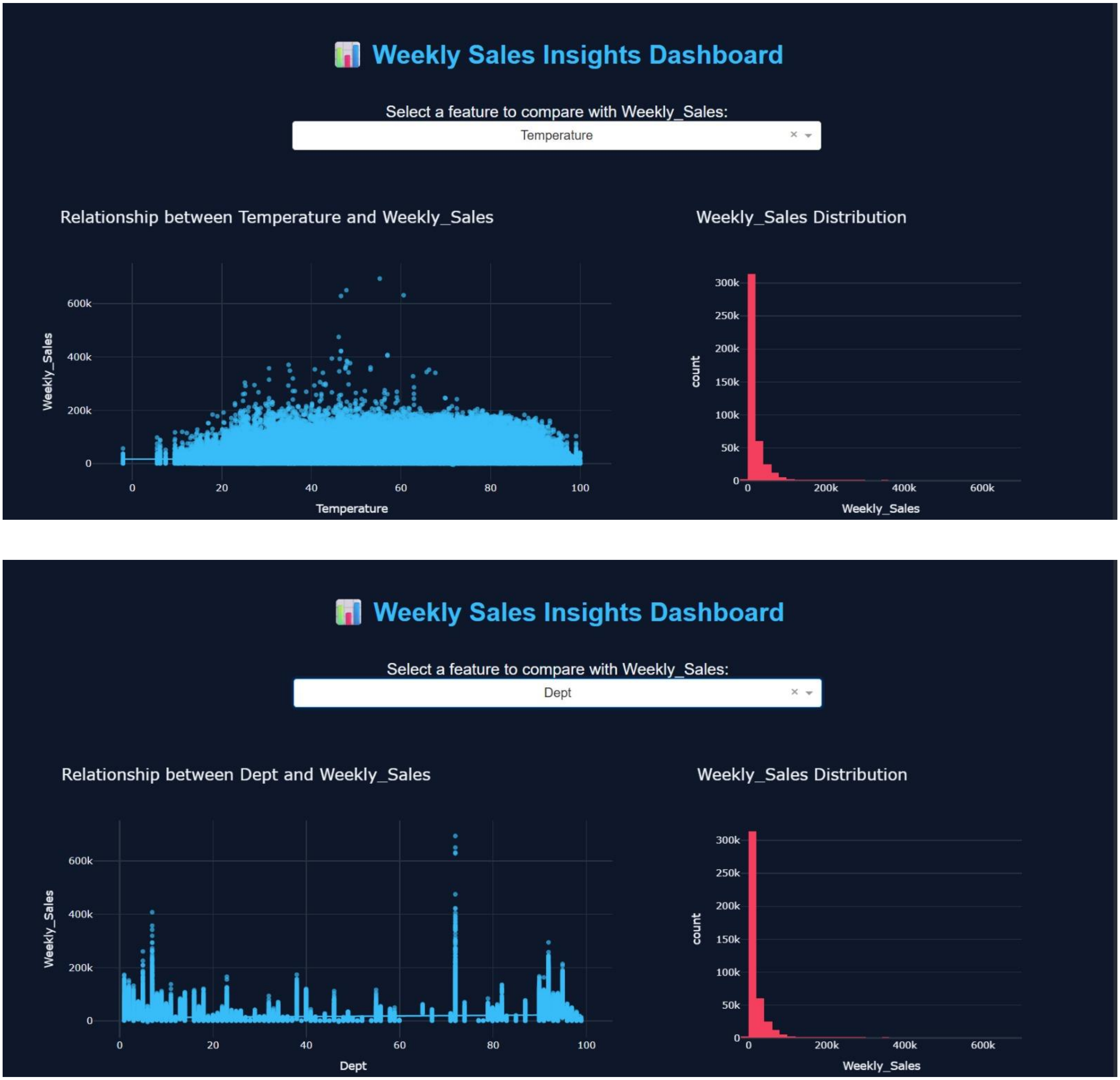
Mean Absolute Error (MAE)

Root Mean Squared Error (RMSE)

R-squared (R^2)



```
MAE: 0.00662586555173845
RMSE: 0.012148465662647101
R2: 0.9980161595521406
```



📊 Weekly Sales Insights Dashboard

Select a feature to compare with Weekly_Sales:

Temperature × ▾

📊 Weekly Sales Insights Dashboard

Select a feature to compare with Weekly_Sales:

Dept × ▾

Fig. 6. Weekly Sales Insights Dashboard

V. Deployment

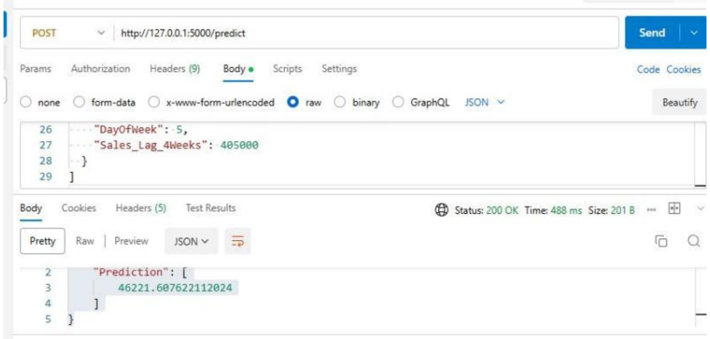
The final stage of the Sales Forecasting and Demand Prediction Project involves deploying the trained model so it can be accessed and used outside the development environment. Deployment transforms the machine learning workflow into a functional application capable of serving real-time or on-demand predictions.

To prepare the model for deployment, the entire preprocessing and prediction steps were combined into a single Pipeline, ensuring that all transformations (such as scaling and encoding) are applied consistently to any new data. This pipeline was then saved as a .pkl file using joblib, allowing efficient loading and execution in production without retraining.

A lightweight Flask application was developed to expose the model as a REST API. The API includes endpoints for health checking and prediction, where users can send JSON input and receive predicted sales values. This design makes it easy to integrate the forecasting system with dashboards, business tools, or other applications.

The Flask app, along with the pipeline .pkl file and necessary dependencies, was deployed on PythonAnywhere. A virtual environment was configured to manage the required libraries, and the WSGI interface was used to serve the application on the cloud.

Link: https://waelalnaqiti5.pythonanywhere.com/



POST http://127.0.0.1:5000/predict

```
    "DayOfWeek": 5,
    "Sales_Lag_4Weeks": 405000
  }
]
```

Status: 200 OK Time: 488 ms Size: 201 B

```
    "Prediction": [
        46221.607622112024
    ]
}
```

VI. Conclusion

The Sales Forecasting and Demand Prediction Project successfully implemented a complete end-to-end machine learning pipeline—from data collection and exploration to model development, tuning, and deployment. By applying advanced preprocessing techniques, exploring multiple algorithms, and optimizing performance through Random Search CV, the project produced a reliable forecasting system capable of handling real-world sales data.

The deployment phase further transformed the model into a practical and accessible tool. Using a unified pipeline saved with joblib and hosted through a Flask API on PythonAnywhere, the system now supports seamless integration with external applications and business workflows.

Overall, this project demonstrates how data-driven forecasting can enhance decision-making, improve operational efficiency, and provide organizations with actionable insights for planning and resource management. The structured approach and scalable deployment ensure that the solution can be extended, updated, and adapted to future business needs