

## PROJECT REPORT

### INTRODUCTION:

#### 1.1 Project Overview:

For firms, sales forecasting is essential to long-term planning and strategic decision-making. Precise forecasts facilitate efficient resource allocation and proactive reaction to market developments for businesses. This research explores the field of sales forecasting, concentrating on Walmart, one of the biggest names in retail.

Walmart is a well-known retailer that runs a series of hypermarkets. It has kindly provided data on 45 stores. This dataset contains monthly sales statistics as well as important store information. Like many other businesses, Walmart plans promotional discount events and carefully schedules them to coincide with important holidays like Christmas, Thanksgiving, Labor Day, and the Super Bowl.

Walmart's method is distinctive in that it emphasizes the possible influence of these holidays on sales by giving the weeks that include these holidays a larger weight during evaluations—five times more than other weeks. Weekly data is provided, which makes it possible to analyze sales trends in relation to certain holidays in great detail.

#### 1.2 Purpose:

This project's main goal is to use machine learning techniques to perform a thorough study of Walmart's sales data. In particular, we want to:

**Recognize Sales Patterns:** Examine past sales information to identify trends, patterns, and seasonality.

**Examine Holiday Effect:** Determine how promotional discounts and the big four holidays—Christmas, Thanksgiving, the Super Bowl, and Labor Day—affect sales.

**Use Machine Learning techniques:** To predict future sales with accuracy, use machine learning techniques such as Random Forest,

Multiple Linear Regressor, Decision Tree, XGBoost, and ARIMA.

Assessing Model Performance Metrics like Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and Mean Absolute Error (MAE) can be used to evaluate the performance and accuracy of each machine learning model.

**Make Flask Integration Active:** Create an interactive model deployment and user-friendly online application with Flask.

By using this diverse approach, we hope to offer insightful information about Walmart's sales dynamics, assisting the retail industry's strategic planning and decision-making processes.

### 2.LITERATURE SURVEY:

#### 2.1 Existing problem:

The retail industry has conducted a great deal of study on sales forecasting since it plays a crucial role in maximizing inventory, resource allocation, and overall business strategy. Accurately forecasting sales is difficult because of a variety of factors, including economic conditions, seasonality, and the influence of promotional activities.

Previous research emphasizes how difficult it is to predict sales in the retail industry and how sophisticated analytical methods are required to take the ever-changing nature of consumer behavior into consideration.

Conventional techniques frequently fail to capture complex patterns, particularly when dealing with abnormalities brought forth by holiday seasons and advertising markdowns.

Algorithms for machine learning have shown promise in improving the accuracy of sales forecasting. In the context of retail, researchers have investigated the use of algorithms such as Random Forest, Decision Trees, XGBoost, and ARIMA, showcasing their capacity to manage non-linear interactions

and identify temporal dependencies in time series data.

## 2.2 References:

1. Md Humayun Kabir, Abdus Sobur, Md Ruhul Amin. Walmart Data Analysis using Machine Learning. International journal of creative research thoughts, Date 7 July 2023:2320-2882.
2. Islam S, Amin S H. Prediction of probable backorder scenarios in the supply chain using Distributed Random Forest and Gradient Boosting Machine learning techniques. Journal of Big Data, 2020, 7: 1-22
3. Singh B, Kumar P, Sharma N, Sharma K P. Sales forecast for amazon sales with time series modeling. In 2020 first international conference on power, control and computing technologies (ICPC2T) 2020: 38-43.

## 2.3 Problem Statement Definition:

The unique difficulties Walmart faces in effectively projecting sales, particularly during promotional markdown events and holidays, have not been fully explored despite breakthroughs in sales forecasting systems. While there is a foundation for understanding basic retail forecasting issues in the literature, a targeted research is necessary due to the particular characteristics of Walmart's dataset.

### Problem Statement:

The present challenge is creating a reliable sales forecasting model that takes into account the weighting of holiday weeks and special markdowns, specifically adapted for Walmart's dataset. This necessitates a thorough investigation of how well machine learning algorithms capture Walmart's sales patterns and a comparison of their performance with conventional forecasting techniques.

In this regard, the study intends to address the following in order to add to the body of current knowledge:

The accuracy of Walmart's sales predictions using machine learning algorithms (XGBoost, Decision Tree, Random Forest, and ARIMA).

The impact of big holidays and promotional markdown events on sales trends.

A comparison between traditional approaches and machine learning models in relation to Walmart's sales forecasting.

The research aims to offer practical insights for improving sales forecasting accuracy in the context of one of the biggest retailers globally by identifying and resolving these particular difficulties.

## 3.IDEATION & PROPOSED SOLUTION

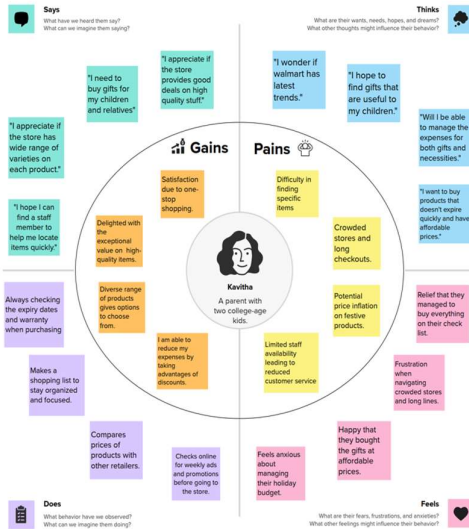
### 3.1 Empathy Map Canvas:

An empathy map is a tool that helps us to gain a deeper understanding of the customers' needs, thoughts, and feelings. The empathy map is not a one-time exercise. It can be revisited and updated as more insights are gathered, allowing us to refine the understanding of the customer over time.

The insights gained from empathy map guide decision-making in marketing strategies and overall customer experience design. It aligns with the principle that successful solutions are those that resonate with the human experience and fulfil genuine user needs. The map is typically divided into four quadrants, each representing a different aspect of the user's experience.

### Reference:

<https://app.mural.co/t/walmart8615/m/walmart8615/1698662602687/1551a2cf388f11c6ff843b8d5ea4bacec62ccc93?sender=u5684e663aeb6db966cda5040>



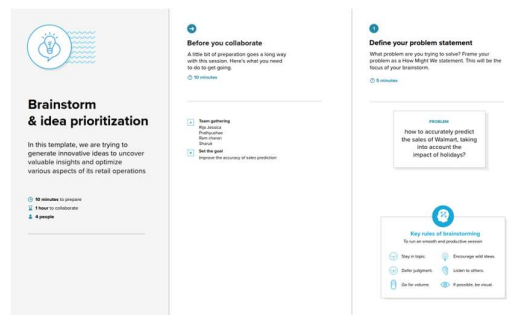
### 3.2 Ideation & Brainstorming:

The objective of brainstorming for Walmart data could be to generate innovative ideas and insights related to improving sales forecasting, enhancing customer experience, optimizing inventory management, or any other relevant business aspect.

Reference:

<https://app.mural.co/t/walmart8615/m/walmart8615/1698904530359/3daad31688cf710edcda862f098cee469daf357c?sender=u948e88338da34e7f8d8d1103>

### Step-1: Team Gathering, Collaboration and Select the Problem Statement



### Step-2: Brainstorm, Idea Listing and Grouping

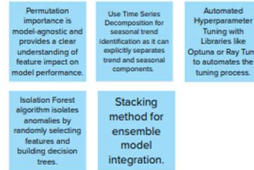
2

### Brainstorm

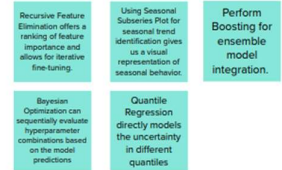
Write down any ideas that come to mind that address your problem statement.

10 minutes

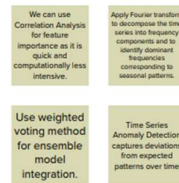
#### Ram Charan



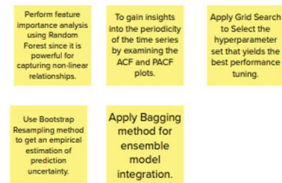
#### Prathyushae



#### Sharuk



#### Jessica



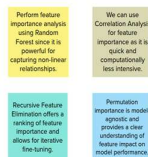
3

### Group ideas

Take turns sharing your ideas while clustering similar or related notes as you go. Once all sticky notes have been grouped, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you can break it up into smaller sub-groups.

20 minutes

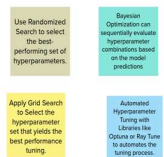
#### Feature Importance



#### Seasonal Trend



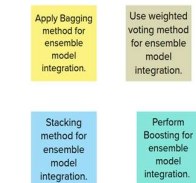
#### Hyperparameter tuning



#### Predictive Uncertainty Analysis



#### Ensemble model Integration



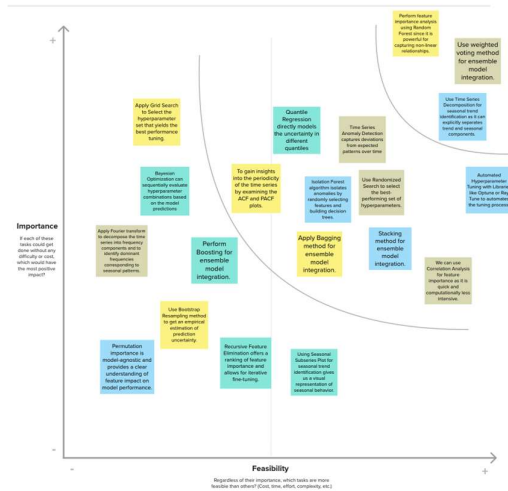
#### Anomaly Detection



### Step-3: Idea Prioritization

20 minutes

**TIP**  
Participants can use their cursors to point at where sticky notes should go on the grid. The facilitator can confirm the spot by using the laser pointer holding the **H** key on the keyboard.



### Proposed Solution Template:

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	<ul style="list-style-type: none"> <li>Accurately predict the weekly sales of Walmart stores taking into account the impact of major holidays.</li> </ul>
2.	Idea / Solution description	<ul style="list-style-type: none"> <li>Implement a model for Walmart to assess the impact of different features on weekly sales, using historical data from 45 stores.</li> </ul>
3.	Novelty / Uniqueness	<ul style="list-style-type: none"> <li>Our solution uniquely integrates advanced machine learning with data analysis, considering not only internal sales data but also external factors like CPI, temperature.</li> </ul>
4.	Social Impact / Customer Satisfaction	<ul style="list-style-type: none"> <li>The Walmart sales predicting machine learning model is expected to enhance social impact and customer satisfaction by optimizing inventory, reducing waste, and tailoring promotions, aligning with the company's commitment to operational efficiency and sustainability.</li> </ul>
5.	Business Model (Revenue Model)	<ul style="list-style-type: none"> <li>Walmart can subscribe to our sales prediction solution, paying a flexible fee depending on how often they want sales predictions. This subscription gives them access to the predictive analytics platform.</li> </ul>
6.	Scalability of the Solution	<ul style="list-style-type: none"> <li>Our solution works for more than just Walmart; it can adapt to the data from different retail companies using flexible machine learning algorithms. This makes it a good fit for various businesses for accurate sales predictions.</li> </ul>

#### 4.REQUIREMENT ANALYSIS:

#### 4.1 Functional Requirements:

The following features and functionalities are outlined in the functional requirements for Walmart's advanced sales forecasting platform:

### Data Integration:

It should be possible to retrieve historical sales data from 45 stores by integrating the system smoothly with Walmart's current data infrastructure.

### Algorithm Selection:

For sales forecasting, users ought to have a selection of algorithms to select from, such as XGBoost, ARIMA, Decision Tree, and Random Forest.

### Holiday Impact Module:

A module on the platform dedicated to analyzing and quantifying the effect of holidays on sales should be included; this module should take into account the weighted evaluation during important holiday weeks.

**Interface User:**

Provide an interface that is simple to use and intuitive so that sales analysts can choose algorithms, change parameters, and evaluate data with ease on the platform.

### Promotional Event Detection:

In order to ensure accurate model training and prediction during these times, implement an automated mechanism to identify promotional markdown events in the sales data.

### Performance Metrics Dashboard:

Provide each forecasting algorithm a dashboard with essential performance data displayed for simple comparison and selection based on recall, accuracy, and precision.

### Integration of Flasks:

Provide an online interface for users to access the forecasting platform by integrating Flask for web deployment.

### Continuous Learning Mechanism :

Add a module that enables the system to adjust and learn from real-time sales data,

increasing the precision of long-term forecasting.

#### 4.2 Non-Functional Requirements:

Non-functional requirements specify the attributes and features the system must have:

##### Performance:

The system should meet Walmart's sales analysts' performance expectations by providing forecasting results in real-time or almost real-time.

##### Scalability:

As Walmart's business grows, the platform should be scalable to accommodate an expanding volume of data and customers.

##### Security:

Put strong security measures in place to safeguard private sales information, guaranteeing its availability, confidentiality, and integrity.

##### Reliability:

It is vital that the forecasting platform exhibits dependability and low downtime to guarantee uninterrupted user accessibility.

##### Usability:

Sales analysts should need little training to operate the platform efficiently because the user interface should be simple to use and intuitive.

##### Compatibility:

Make sure the platform is compatible with a range of web browsers and devices to give users options in how they can access it.

##### Maintainability:

Consider the ease of maintenance when designing the system to enable upgrades, bug repairs, and algorithm improvements.

##### Compliance:

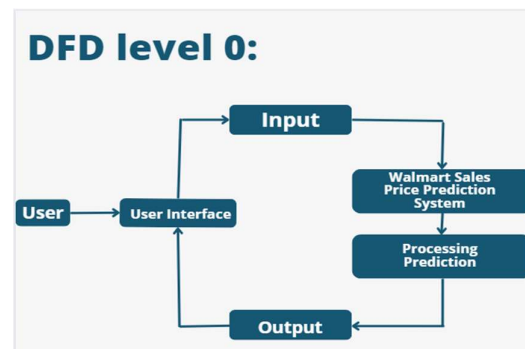
Verify that the platform conforms with industry standards and applicable data protection laws.

The enhanced sales forecasting platform for Walmart is being developed and implemented on the basis of these functional and non-functional requirements. They direct the development team in order to produce a solution that satisfies the particular requirements and demands of Walmart's sales analysts.

## 5. PROJECT DESIGN:

### 5.1 Data Flow Diagrams & User Stories:

Data Flow Diagram (DFD) for a Walmart sales price prediction system involves illustrating the flow of data between different components or processes. In this case, we consider the point of view of a user interacting with the system.



1. User goes to user interface in the website.

2. User will give the input required for prediction.

3. Walmart Sales Price Prediction System, a machine learning model will take the user input.

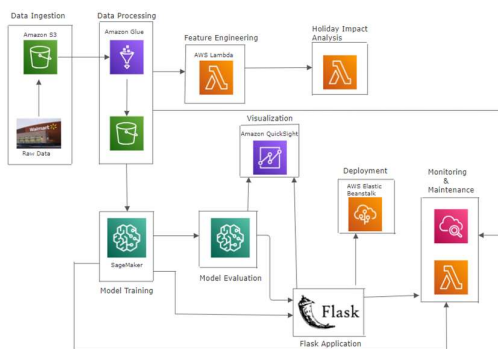
4. Processing and Prediction takes place in Walmart Sales Price Prediction System.

5. User will receive the predicted output from the Walmart Sales Price Prediction System.

## User Stories:

User Type	Functional Requirement (Epic)	User Story Number	User Story/Task	Acceptance criteria	Priority	Release
Customer (Web User)	Input	USN-1	As a user, I can easily input the information in this user-friendly interface.	I can access the user interface.	High	Sprint-1
Customer (Mobile User)	Dashboard	USN-2	As a user, I can interact with the website on mobile phones without any display restrictions.	I can access this website in smartphones and tablets.	High	Sprint-3
Customer (New User)	Interface	USN-3	As a user, I can easily get to know what inputs to enter in each field.	I can get the clear and concise instructions within the input interface.	Medium	Sprint-1
Customer	Prediction	USN-4	As a user, I can get detailed insights about the predicted sales.	I can get to know whether the predicted sales are high, low or moderate compared to historical sales.	Low	Sprint-2

## 5.2 Solution Architecture:

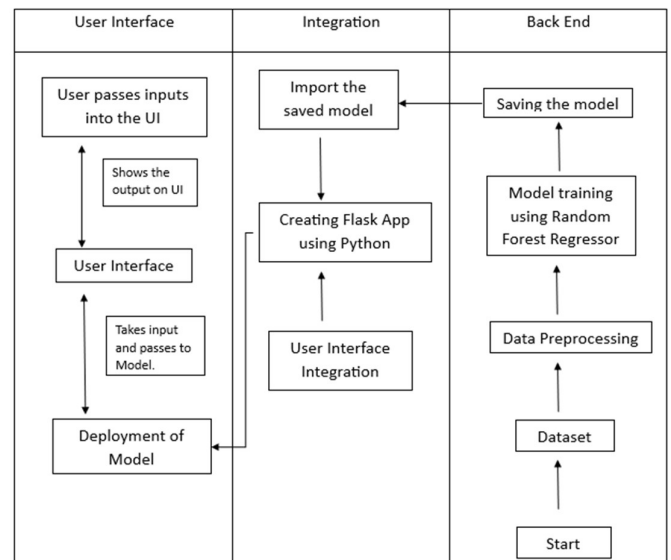


- First, the raw data from Walmart stores is stored in Amazon S3.
- Then we use Amazon Glue to perform ETL operations and store the processed data into another S3 bucket.
- Next, with the help of AWS Lambda, we perform feature engineering to extract and engineer features from the processed data and also to compute the impact weights of holidays.
- Then we use Amazon SageMaker for training machine learning models.
- Then evaluate the models using Amazon SageMaker

- Develop the flask application for user interaction
- Flask application is then deployed on AWS Elastic Beanstalk
- Then we use QuickSight, a visualization tool to visualize and to gain insights from the data.
- Aws CloudWatch and AWS Lambda functions are set up to monitor the deployed models and infrastructure.

## 6.PROJECT PLANNING SCHEDULING:

### 6.1 Technical Architecture:



### 6.2 Sprint Planning & Estimation:

Sprints	Functional Requirement (Epic)	User Story Number	User Story/Task	Story Points	Priority	Team Members
Sprint-1	Input	USN-1	As a user, I can easily input the information in this user-friendly interface.	3	High	Jessica Sharuk
Sprint-3	Dashboard	USN-2	As a user, I can interact with the website on mobile phones without any display restrictions.	5	High	Jessica Prathyushae
Sprint-1	Interface	USN-3	As a user, I can easily get to know what inputs to enter in each field.	2	Medium	Sharuk Ram Charan
Sprint-2	Prediction	USN-4	As a user, I can get detailed insights about the predicted sales.	5	High	Prathyushae Ram Charan

### 6.3 Sprint Delivery Schedule:

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	7	1 Days	20 Nov 2023	20 Nov 2023	7	20 Nov 2023
Sprint-2	8	1 Days	21 Nov 2023	21 Nov 2023	8	21 Nov 2023
Sprint-3	5	1 Days	22 Nov 2023	22 Nov 2023	5	22 Nov 2023

## 7.CODING & SOLUTION:

### 7.1 Feature 1: Weak of the year

This would capture almost all the lag and lead effects of holidays

```
df['Date']=pd.to_datetime(df['Date'])
```



```
df['Month']=df['Date'].dt.month
df['Year']=df['Date'].dt.year
df['Week'] = df['Date'].dt.isocalendar().week
df['Day_of_week']=df['Date'].dt.day_name()
```

## 8.PERFORMANCE TESTING:

```
print("MSE:", metrics.mean_squared_error(y_test,y_pred))
MSE: 12083384.617536364

print('RMSE:', np.sqrt(metrics.mean_squared_error(y_test,y_pred)))
RMSE: 3476.116312429198

print('MAE:',metrics.mean_absolute_error(y_test,y_pred))
MAE: 1417.3229688475956

from sklearn.model_selection import TimeSeriesSplit,cross_val_score
cv=TimeSeriesSplit(n_splits=10)
scores = cross_val_score(rf, x, y, cv = cv)
print("Cross Validation Scores: ", scores)
Cross Validation Scores: [0.36581115 0.46111931 0.69842702 0.61893673 0.71898625 0.71576812
0.55508449 0.56221755 0.73802908 0.84961531]

np.mean(scores)
0.6283915011292479
```

## 9.RESULT:

### 9.1 Output Screenshots:

Predicted weekly sales for Store 1, Department 1 in the month of November is : 32118.43

The predicted sales are within a moderate range.

- dept: 0.2
- size: 0.15
- store: 0.12
- Week: 0.1
- CPI: 0.08
- temp: 0.07
- b: 0.06
- Month: 0.05
- a: 0.04
- IsHoliday: 0.03
- Year: 0.02
- c: 0.01

Most significant feature in the predicted sale is dept

## 10.ADVANTAGES & DISADVANTAGES:

### 10.1 Advantages:

#### 1. Accurate Sales forecasts:

By utilizing cutting-edge machine learning algorithms, the system offers precise sales

projections that support well-informed decision-making.

#### 2. Holiday Impact Analysis:

Strategic planning regarding promotional markdown events is made possible by an effective examination of the impact of holidays on sales.

#### 3. Multiple Algorithm Support:

The availability of multiple algorithms (Random Forest, Decision Tree, XgBoost, and ARIMA) offers forecasting alternatives and flexibility.

#### 4. User-Friendly Interface:

Users with different degrees of technical competence can utilize the system because of its user-friendly interface.

### 10.2 Disadvantages:

#### 1.Dependency on Historical Data:

The system may encounter difficulties in unpredictable market conditions because accurate forecasting is mostly dependent on historical sales data.

#### 2.Algorithm Sensitivity:

Forecast accuracy may be impacted by suboptimal settings because machine learning algorithms' performance is sensitive to parameter adjustment.

#### 3.Resource Intensiveness:

Requiring strong computational skills, some machine learning methods may be resource-intensive.

## 11.CONCLUSION:

By using machine learning techniques, the Walmart Sales Forecasting System effectively tackles the problem of projecting future sales. Walmart is able to effectively arrange promotional events and maximize sales during crucial periods by analyzing the effects of holidays. The support for several algorithms

guarantees flexibility in selecting the best model for forecasting.

The benefits of the system, like its user-friendly interface and precise projections, add to its efficacy. Nonetheless, consideration should be given to issues with algorithm sensitivity and data dependency.

## 12. FUTURE SCOPE:

### 1.Integration of External Factors:

For a more thorough study, expand the system to incorporate external factors (such as industry trends and economic data).

### 2. Real-time Data Processing:

Investigate your alternatives for processing data in real-time to quickly adjust to changing market conditions.

### 3. Enhanced User Customization:

Give users additional opportunities to customize parameters and algorithms to fit forecasts to particular store needs.

## 13. APPENDIX:

### App2.py

```
import numpy as np

from numpy.core.fromnumeric import size

import pandas as pd

from sklearn.model_selection import
train_test_split

from flask import Flask, render_template,
request

import pickle

import datetime as dt

import calendar

import os

app = Flask(__name__)

loaded_model =
pickle.load(open('rf_model.pkl','rb'))
```

```
fet = pd.read_csv('merged_data.csv')

feature_importance_dict = {

    'dept': 0.2,

    'size': 0.15,

    'store': 0.12,

    'Week': 0.1,

    'CPI': 0.08,

    'temp': 0.07,

    'b': 0.06,

    'Month': 0.05,

    'a': 0.04,

    'IsHoliday': 0.03,

    'Year': 0.02,

    'c': 0.01

}

@app.route('/')

def home():

    return render_template('index1.html')

@app.route('/predict', methods=['POST'])

def predict():

    store = request.form.get('store')

    dept = request.form.get('dept')

    size=request.form.get('size')

    temp=request.form.get('temp')

    a=request.form['aRadio']

    b=request.form['bRadio']

    c=request.form['cRadio']

    CPI=request.form.get('CPI')

    Unemployment=request.form.get('Unemploy
ment')

    IsHoliday = request.form['isHolidayRadio']
```



```

date = request.form.get('date')

d=dt.datetime.strptime(date, '%Y-%m-%d')

Month = d.month

Year = (d.year)

Week=d.isocalendar().week

month_name=calendar.month_name[Month]

print("year = ", type(Year))

print("year val = ", Year, type(Year), Month)

X_test = pd.DataFrame({'Store': [store],
'Dept': [dept], 'IsHoliday': [IsHoliday],
                        'Temperature': [temp],
                        'CPI': [CPI], 'Size': [size],
                        'Month': [Month],
                        'Year': [Year], 'Week': [Week],
                        'A': [a], 'B': [b], 'C': [c]})

print("X_test = ", X_test.head())

print("type of X_test = ", type(X_test))

print("predict = ", store, dept, date,
IsHoliday)

y_pred = loaded_model.predict(X_test)

output=round(y_pred[0],2)

print("predicted = ", output)

insights = generate_insights(output)

most_significant_feature =
max(feature_importance_dict, key=feature_im
portance_dict.get)

print("significant_feature=", most_significant_f
eature)

return render_template('index1.html',
output=output, store=store, dept=dept,
                        month_name=month_name,
Year=Year,
insights=insights, feature_importance=feature
_importance_dict,

```

```

most_significant_feature=most_significant_fe
ature)

def generate_insights(output):

    insights = "No specific insights available."

    if output > 50000:

        insights = "The predicted sales are high.
Consider potential factors contributing to this
peak."

    elif output < 5000:

        insights = "The predicted sales are
relatively low. Evaluate potential reasons for
the decrease."

    else:

        insights = "The predicted sales are within
a moderate range."

    return insights

if __name__ == "__main__":

    app.run(debug=False)

```

### index1.html

```

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport"
content="width=device-width, initial-
scale=1.0">

    <title>Walmart Sales Prediction</title>

    <!-- Add Bootstrap CSS link here -->

    <link
href="https://cdn.jsdelivr.net/npm/bootstrap
@5.0.2/dist/css/bootstrap.min.css"
rel="stylesheet" integrity="sha384-
EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztc
QTWfSpd3yD65VohhpuuCOmLASjC"
crossorigin="anonymous">

```

```

<style>
  body {
    width: 100%;
    height: 100%;
    background-image:
url('static/images/background1.png');
    background-size: cover;
    background-position: center top;
    background-repeat: no-repeat;
    color: white;
    margin: 0;
  }
  .container {
    background-color: rgba(0, 0, 0, 0.6);
    margin-top: 50px;
  }
</style>
<body>
  <h1 class="text-center" style="margin-
top:70px">Walmart Sales Prediction</h1>
  <div class="container" >
    <form action="/predict" method="post">
      <div class="row align-items-start ">
        <div class="col">
          <label for="store" class="col-form-
label col-form-label-lg">Store</label>
          <small class="form-text text-
muted">(Store Number)</small>
          <input type="text" name="store"
class="form-control form-control-sm"
placeholder="Store (e.g 1,2,3)" id="store">
        </div>
        <div class="col">

```

```

          <label for="dept" class="col-form-
label col-form-label-lg">Department</label>
          <small class="form-text text-
muted">(Department Number)</small>
          <input type="number" name="dept"
class="form-control form-control-sm"
placeholder="Dept (e.g 1,2,3)" id="dept">
        </div>
      </div>
      <div class="col">
        <label for="size" class="col-form-
label col-form-label-lg">Size</label>
        <small class="form-text text-
muted">(Size of the store)</small>
        <input type="number" name="size"
class="form-control form-control-sm"
id="size" placeholder="Size">
      </div>
    </div>
    <div class="row align-items-center">
      <div class="col">
        <label for="temp" style="margin-
top:5px" class="col-form-label col-form-label-
lg">Temperature</label>
        <small class="form-text text-
muted">(Average temperature in the
region)</small>
        <input type="number"
name="temp" class="form-control form-
control-sm" id="temp"
placeholder="Temperature" >
      </div>
      <div class="col">
        <label for="date" style="margin-
top:5px" class="col-form-label col-form-label-
lg">Date:</label>
        <small class="form-text text-
muted">(DD-MM-YYYY)</small>

```

```

        <input type="date" id="date"
class="form-control form-control-sm"
name="date" >

    </div>

    <div class="col">

        <label for="CPI" style="margin-
top:5px" class="col-form-label col-form-label-
lg">CPI</label>

        <small class="form-text text-
muted">(Consumer Price Index)</small>

        <input type="number" name="CPI"
class="form-control form-control-sm"
placeholder="CPI (Round-Up the decimals)"
id="CPI" >

    </div>
</div>

<div class="row align-items-end">

    <div class="col">

        <label for="isholiday" style="margin-
top:5px" class="col-form-label col-form-label-
lg">IsHoliday:</label>

        <small class="form-text text-
muted">(special holiday week)</small>

        <input class="form-check-input"
type="radio" value="1"
name="isHolidayRadio" id="yes"
style="margin-top:25px">

        <label for="yes" style="margin-
top:5px" class="col-form-label col-form-label-
lg" > Yes </label>

        <input class="form-check-input"
type="radio" value="0"
name="isHolidayRadio" id="no"
style="margin-top:25px">

        <label for="no" style="margin-
top:5px" class="col-form-label col-form-label-
lg"> No </label>

    <br>

```

```

    </div>

    <div class="col">

        <label for="a" style="margin-top:5px"
class="col-form-label col-form-label-lg" >Type-
A</label>

        <small class="form-text text-
muted">(Type of the store)</small>

        <input class="form-check-input"
type="radio" value="1" name="aRadio"
id="yes" style="margin-top:25px">

        <label for="yes" style="margin-
top:5px" class="col-form-label col-form-label-
lg"> Yes </label>

        <input class="form-check-input"
type="radio" value="0" name="aRadio"
id="no" style="margin-top:25px">

        <label for="no" style="margin-
top:5px" class="col-form-label col-form-label-
lg"> No </label>

    <br>
</div>

    <div class="col">

        <label for="b" style="margin-top:5px"
class="col-form-label col-form-label-lg">Type-
B</label>

        <small class="form-text text-
muted">(Type of the store)</small>

        <input class="form-check-input"
type="radio" value="1" name="bRadio"
id="yes" style="margin-top:25px">

        <label for="yes" style="margin-
top:5px" class="col-form-label col-form-label-
lg"> Yes </label>

        <input class="form-check-input"
type="radio" value="0" name="bRadio"
id="no" style="margin-top:25px">

        <label for="no" style="margin-
top:5px" class="col-form-label col-form-label-
lg"> No </label>

```

```

    </div>
</div>

<div class="row align-items-end">

    <div class="col">

        <label for="c" style="margin-top:5px"
class="col-form-label col-form-label-lg">Type-
C</label>

        <small class="form-text text-
muted">(Type of the store)</small>

        <input class="form-check-input"
type="radio" value="1" name="cRadio"
id="yes" style="margin-top:25px">

        <label for="yes" style="margin-
top:5px" class="col-form-label col-form-label-
lg"> Yes </label>

        <input class="form-check-input"
type="radio" value="0" name="cRadio"
id="no" style="margin-top:25px">

        <label for="no" style="margin-
top:5px" class="col-form-label col-form-label-
lg"> No </label>

        <br>

    </div>

    <div class="col">

        <button type="submit" style="margin-
top:5px" class="btn btn-primary">Predict
Sales</button>

    </div>

</div>

</form>

{% if output %}

    <div style="font-size: x-large; color:
ghostwhite; font-family: serif; margin-top:
5%;">

        <p id="text">Predicted weekly sales for
Store {{store}}, Department {{dept}} in the

```

```

month of {{month_name}} {{year}} is : {{
output }} </p>

    <p id="text">{{insights}}</p>

    <ul>

        {% for feature, importance in
feature_importance.items() %}

        <li>{{ feature }}: {{ importance }}</li>

        {% endfor %}

    </ul>

    <p id="text">Most significant feature in
the predicted sale is {{
most_significant_feature }}</p>

</div>

{% endif %}

</div>

<script
src="https://code.jquery.com/jquery-
3.3.1.slim.min.js" integrity="sha384-
q8i/X+965DzO0rT7abK41JStQIAqVgRVzpbzo5s
mXKp4YfRvH+8abtTE1Pi6jizo"
crossorigin="anonymous"></script>

<script
src="https://cdn.jsdelivr.net/npm/popper.js/1.14.7/umd/popper.min.js"
integrity="sha384-
Uo3fC0Mzve8Wn8E8+1n5/wq4FF02xIM5ibzq1
Oi6jFZo5zK7u2EC1gM5S5hMVoVk"
crossorigin="anonymous"></script>

<script
src="https://stackpath.bootstrapcdn.com/boo
tstrap/4.3.1/js/bootstrap.min.js"
integrity="sha384-
JjSmVgyd0p3pXB1rRibZUAYoIlly6OrQ6VrjIEaFf/
nJGzlxFDsf4x0xIM+B07jRM"
crossorigin="anonymous"></script>

</body>

</html>

```

