FORECASTING SALES OF STORE USING IBM WATSON STUDIO

Mini Project Report

Submitted By **(BATCH NO: CSE_015)**

GOWROJU LAXMIPRIYA	**18UK1A0518**
CHALLAKONDA SRICHANDAN	**18UK1A0567**
JUNNUTHULA AKSHAYA	**18UK1A0523**
MANDA MANASA	**18UK1A0531**

INDEX

ABSTRACT

Sales forecasting is an essential task for the management of a store. Being able to estimate the number of products that a retail store is going to sell in the future will allow the owners of these shops to prepare the inventory that they will need.

In this project, we are building a system that analyses the previous trends of sales which includes sales on various days and predicts future sales. The goal of this project is to forecast the sales of stores by using time series analysis. Here time series analysis algorithms such as RNN (Recurrent Neural Network) & LSTM (Long Term Short Memory) are used to analyze the past trends of sales of stores. Create and deploy flask-based web Application and integrate AI model to it.

The objective of the project is to build a web application where the user gives the last ten days' sales values and gets the prediction for the 11th day which is showcased on UI.

1. INTRODUCTION

1.1 OVERVIEW

Sales forecasting is the process of using a company's sales records over the past year's or past days to predict the short term or long terms sales performance in the future . This is one of the pillars of proper financial planning.

One of the most common yet basic challenges that the management of companies face in making business sales forecasts is that their usual approach is a "top to down "one . This approach leaves very little scope for interaction with the sales manager and the salespersons during the data collection process.

Sales forecasting is an essential task for the management of a store. Being able to estimate the number of products that a retail store is going to sell in the future will allow the owners of these shops to prepare the inventory that they will need.

In this project, we are building a system that analyses the previous trends of sales which includes sales on various days and predicts future sales. The goal of this project is to forecast the sales of stores by using time series analysis. Here time series analysis algorithms such as RNN (Recurrent Neural Network) & LSTM (Long Term Short Memory) are used to analyze the past trends of sales of stores. Create and deploy flask-based web Application and integrate AI model to it.

The objective of the project is to build a web application where the user gives the last ten days' sales values and gets the prediction for the 11th day which is showcased on UI.

1.2 PURPOSE

- Understand the historical demand for the products in the last 11 days.
- Analyse the 11^{th} day sales of the store.
- Use the time series forecasting.
- Create a web based UI to produce forecasts.

2. LITERATURE SURVEY

2.1 EXISTING PROBLEM

We know that, for every store, sale is one of the most important factors as it is directly proportionate with store profits and losses.

Hence most store owners are interested in forecast of the sales of stores .Previously, it used to be done by manual processes using balance sheets and spread sheets.

Sales forecast team used to study and analyze old statistics and reports and by certain mathematical calculations and through some complex algorithms they try to find out future predictive sale of a store or a company.

This type of manual approach could not able to provide desired exact accuracy and mostly it is fail during stating quarters. To check accuracy , higher management has to wait for last quarter sales and then only they can able to track whatever report generated by sale forecast team is accurate or not . Also it is a quite leanly process.

2.2 PROPOSED SOLUTION

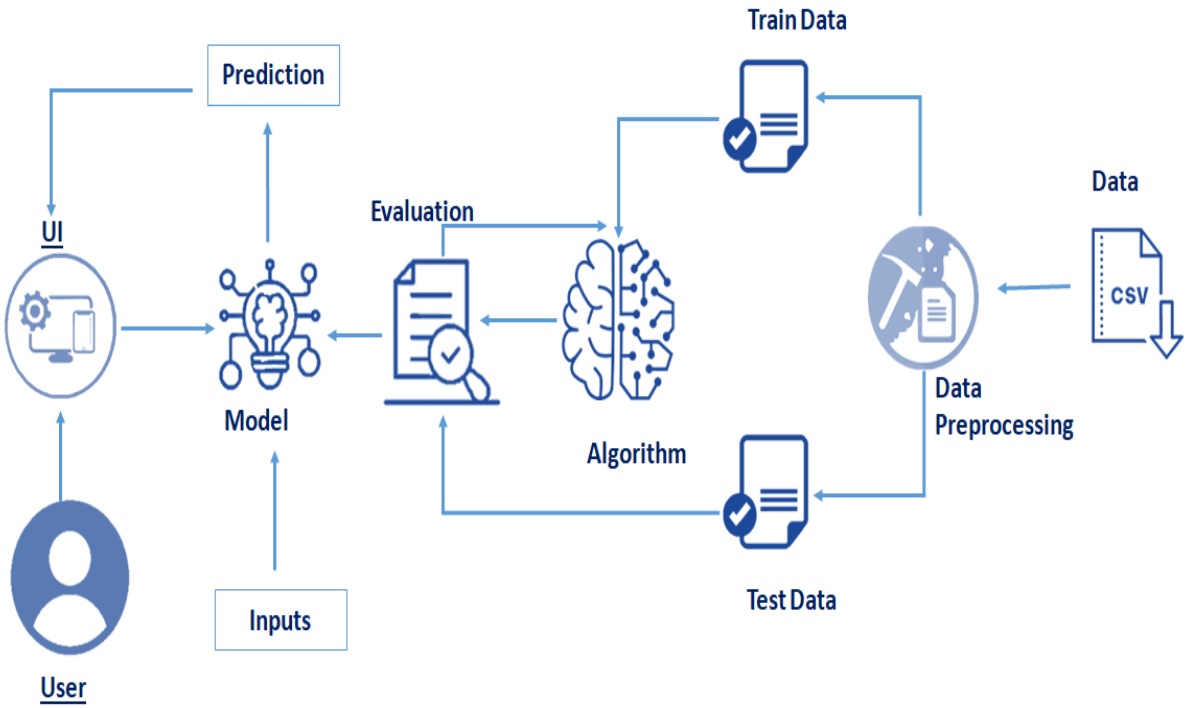
To overcome the drawbacks of manual sales forecast team's results, we have

Proposed one deep learning based sales forecasting system. This system can able to provide more than 95% accuracy in the sales forecasting of a store or a company.

This system can able to generate whole sales reports with requirement of necessary manpower and other resources. It means that by using this sales forecasting system, one can easily find out that to achieve particular sales target , how much manpower and how much resources are required.

3. THEORETICAL ANALYSIS

3.1 BLOCK DIAGRAM



3.2 HARDWARE/SOFTWARE DESIGNING

To complete this project you should have the following software and packages.

- Anaconda Navigator
  Jupyter notebook
  Spyder IDE
- Tensor flow
- Keras
- Notepad

4. PROJECT FLOW

To accomplish this, we have to complete all the activities and tasks listed below

- Data Collection.
  - Collect the dataset or Create the dataset

Data Preprocessing.

- Import the Libraries.
- Importing the dataset.
- Analyze the data
- Taking care of Missing Data
- Feature Scaling
- Data Visualization
- Splitting Data into Train and Test.
- Creating a dataset with a sliding window.

- Model Building
  - Import the model building Libraries
  - Initializing the model
  - Adding LSTM Layers
  - Adding Output Layer
  - Configure the Learning Process
  - Training the model
  - Model Evaluation
  - Save the Model
  - Test the Model

Application Building

- Create an HTML file
- Build Python Code

5. DATA

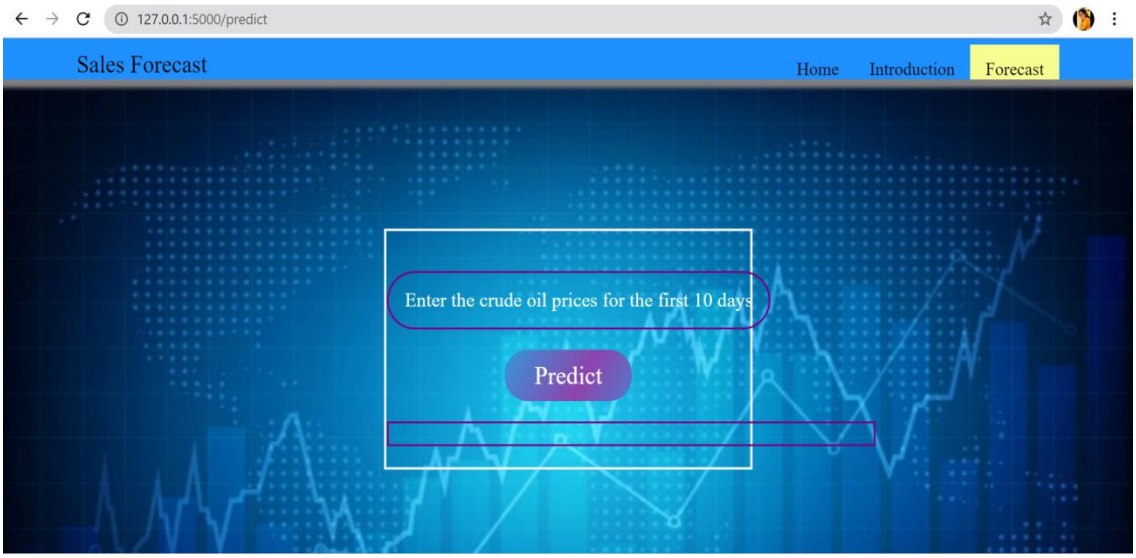
Deep Learning depends heavily on data, without data, a machine can't learn. It is the most crucial aspect that makes neural network training possible. In Deep Learning projects, we need a training data set. It is the actual data set used to train the model for performing various actions.

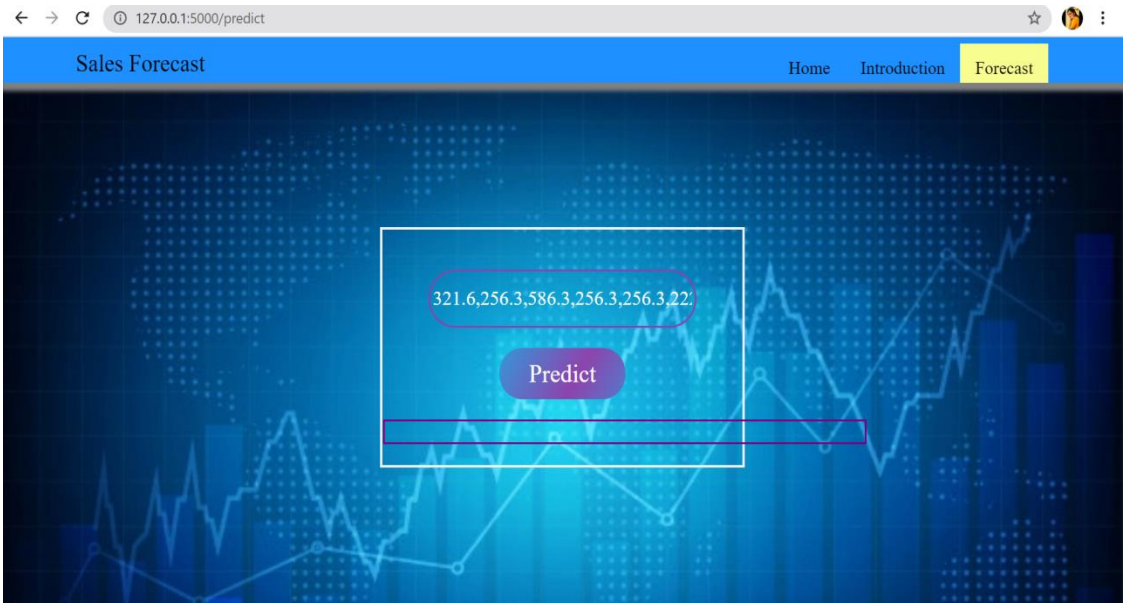
- Our Sales dataset contains

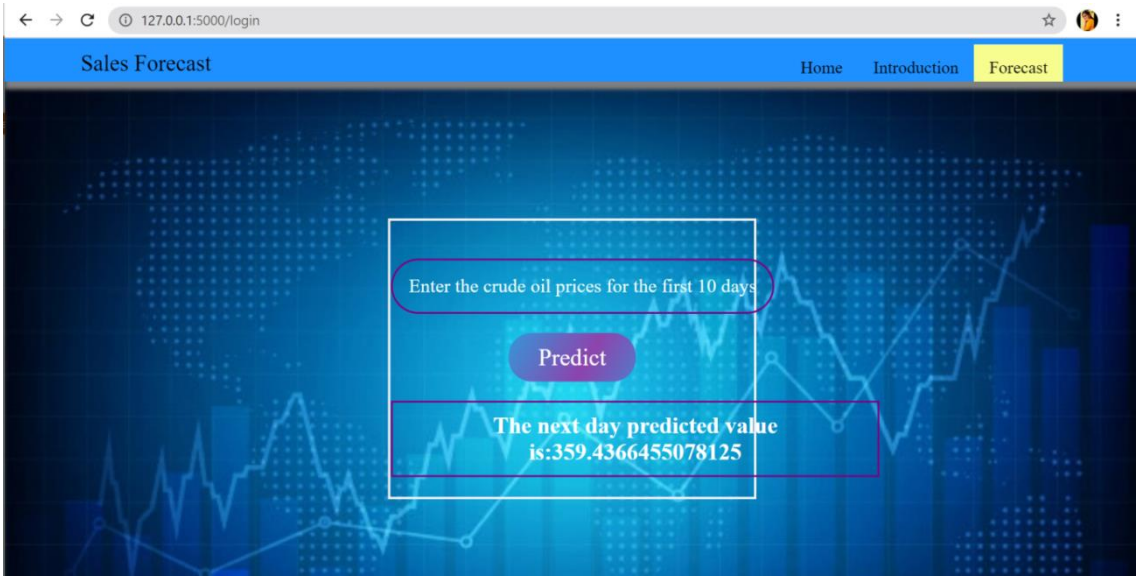
  1. HQ - This attribute represents HeadQuarters of the company.
  2. Country - This attribute represents the country of the outlet.
  3. State_of_outlet - This attribute represents the state of the outlet.
  4. City_of_outlet - This attribute represents the city of the outlet.
  5. Month - This attribute represents the month(ranging from 1 to 12)
  6. Day - This attribute represents a day of the month(ranging from 1 to 31)
  7. Year - This attribute represents the year(ranging from 2005 to 2016)
  8. Total_Sales - Target variable (total sales on a particular day). It contains total sales from 19886 to 2018.

	A	B	C	D	E	F	G	H
1	HQ	Country	State_of_outlet	City_of_outlet	Month	Day	Year	Total_Sales
2	Asia	India		Bombay (Mumba	1	1	2005	72.2
3	Asia	India		Bombay (Mumba	1	2	2005	72.7
4	Asia	India		Bombay (Mumba	1	3	2005	74.3
5	Asia	India		Bombay (Mumba	1	4	2005	78.9
6	Asia	India		Bombay (Mumba	1	5	2005	81.5
7	Asia	India		Bombay (Mumba	1	6	2005	78.8
8	Asia	India		Bombay (Mumba	1	7	2005	77.3
9	Asia	India		Bombay (Mumba	1	8	2005	79.3
10	Asia	India		Bombay (Mumba	1	9	2005	78.8
11	Asia	India		Bombay (Mumba	1	10	2005	76.6
12	Asia	India		Bombay (Mumba	1	11	2005	76.7
13	Asia	India		Bombay (Mumba	1	12	2005	76.7
14	Asia	India		Bombay (Mumba	1	13	2005	78.8
15	Asia	India		Bombay (Mumba	1	14	2005	76.1
16	Asia	India		Bombay (Mumba	1	15	2005	77.1
17	Asia	India		Bombay (Mumba	1	16	2005	75.5
18	Asia	India		Bombay (Mumba	1	17	2005	74.8
19	Asia	India		Bombay (Mumba	1	18	2005	71.2
20	Asia	India		Bombay (Mumba	1	19	2005	72.1
21	Asia	India		Bombay (Mumba	1	20	2005	76.8
22	Asia	India		Bombay (Mumba	1	21	2005	78
23	Asia	India		Bombay (Mumba	1	22	2005	74.9
24	Asia	India		Bombay (Mumba	1	23	2005	72.6
25	Asia	India		Bombay (Mumba	1	24	2005	73.5
26	Asia	India		Bombay (Mumba	1	25	2005	73.6
27	Asia	India		Bombay (Mumba	1	26	2005	72.8
28	Asia	India		Bombay (Mumba	1	27	2005	75.8
29	Asia	India		Bombay (Mumba	1	28	2005	76
30	Asia	India		Bombay (Mumba	1	29	2005	75.9
31	Asia	India		Bombay (Mumba	1	30	2005	75.7
32	Asia	India		Bombay (Mumba	1	31	2005	76.4
33	Asia	India		Bombay (Mumba	2	1	2005	78.2
34	Asia	India		Bombay (Mumba	2	2	2005	79.6
35	Asia	India		Bombay (Mumba	2	3	2005	80.2
36	Asia	India		Bombay (Mumba	2	4	2005	81.6
37	Asia	India		Bombay (Mumba	2	5	2005	81
38	Asia	India		Bombay (Mumba	2	6	2005	80.1
39	Asia	India		Bombay (Mumba	2	7	2005	80.6
40	Asia	India		Bombay (Mumba	2	8	2005	81.6
41	Asia	India		Bombay (Mumba	2	9	2005	83.8
42	Asia	India		Bombay (Mumba	2	10	2005	82.3
43	Asia	India		Bombay (Mumba	2	11	2005	80.4
44	Asia	India		Bombay (Mumba	2	12	2005	82.7
45	Asia	India		Bombay (Mumba	2	13	2005	85.3
46	Asia	India		Bombay (Mumba	2	14	2005	83.8
47	Asia	India		Bombay (Mumba	2	15	2005	81.1
48	Asia	India		Bombay (Mumba	2	16	2005	81.3
49	Asia	India		Bombay (Mumba	2	17	2005	79
50	Asia	India		Bombay (Mumba	2	18	2005	77.3
51	Asia	India		Bombay (Mumba	2	19	2005	75.9
52	Asia	India		Bombay (Mumba	2	20	2005	72.6
53	Asia	India		Bombay (Mumba	2	21	2005	77.3
54	Asia	India		Bombay (Mumba	2	22	2005	77.4
55	Asia	India		Bombay (Mumba	2	23	2005	75.2
56	Asia	India		Bombay (Mumba	2	24	2005	77.8
57	Asia	India		Bombay (Mumba	2	25	2005	81.2
58	Asia	India		Bombay (Mumba	2	26	2005	85.6
59	Asia	India		Bombay (Mumba	2	27	2005	83
60	Asia	India		Bombay (Mumba	2	28	2005	79.6
61	Asia	India		Bombay (Mumba	3	1	2005	79.9
62	Asia	India		Bombay (Mumba	3	2	2005	79.2
63	Asia	India		Bombay (Mumba	3	3	2005	79.2
64	Asia	India		Bombay (Mumba	3	4	2005	78.4
65	Asia	India		Bombay (Mumba	3	5	2005	78.3

6. RESULTS


127.0.0.1:5000/predict
Sales Forecast
Home
Introduction
Forecast
Enter the crude oil prices for the first 10 days
Predict


127.0.0.1:5000/predict
Sales Forecast
Home
Introduction
Forecast
321.6,256.3,586.3,256.3,256.3,22
Predict


127.0.0.1:5000/login
Sales Forecast
Home
Introduction
Forecast
Enter the crude oil prices for the first 10 days
Predict
The next day predicted value
is:359.4366455078125

7. APPLICATIONS

- Forecasting has applications in a wide range of fields where estimates of future conditions are useful. Not everything can be forecast reliably, if the factors that relate to what is being forecast are known and well understood and there is a significant amount of data that can be used very reliable forecasts can often be obtained. If this is not the case or if the actual outcome is affected by the forecasts, the reliability of the forecasts can be significantly lower.

- Forecasting has also been used to predict the development of conflict situations.

8. CONCLUSION

- In this project we have done the prediction of 11^{th} day sales of a store after entering the before 10 days sales data
- Prediction is done using LSTM and RNN algorithms

9. FUTURE SCOPE

Sales forecasting is an essential task for the management of a store. Being able to estimate the number of products that a retail store is going to sell in the future will allow the owners of these shops to prepare the inventory that they will need.

10. BIBLIOGRAPHY

- https://www.geeksforgeeks.org/
- https://www.w3schools.com/
- https://stackoverflow.com/
- https://www.lucidchart.com/pages/
- https://smartinternz.com/Student/guided_project_info/4652#

11. APPENDIX

intro.html

```
<html>
<style>
.header {       position: relative;
                top:0;
                margin:0px;
                z-index: 1;
                left: 0px;
                right: 0px;
                position: fixed;
                background-color: #1E90FF ;
                color: white;
                box-shadow: 0px 8px 4px #FAFAFA;
                overflow: hidden;
                padding-left:20px;
                font-family: 'Josefin Sans';
                font-size: 2vw;
                width: 100%;
                height:8%;
                text-align: center;
        }
        .topnav {
 overflow: hidden;
```

```
  background-color: #ECFAC0;
}
.topnav-right a {
  float: left;
  color: black;
  text-align: center;
  padding: 14px 16px;
  text-decoration: none;
  font-size: 18px;
}
.topnav-right a:hover {
  background-color: #ECFAC0;
  color: black;
}
.topnav-right a.active {
  background-color: #ECFAC0
  color: white;
}
.topnav-right {
  float: right;
  padding-right:100px;
}
body {
  background-color: ;
  background-repeat: no-repeat;
```

```
  background: #310404
url(https://i.pinimg.com/originals/2e/e6/99/2ee6998e34c3e2eff7b894c66cfc5267.jpg)
no-repeat centercenter fixed;

  background-size:cover;

  background-position: 0px 0px;

  }

  .button {

  background-color: #091425;

  border: none;

  color: white;

  padding: 15px 32px;

  text-align: center;

  text-decoration: none;

  display: inline-block;

  font-size: 12px;

  border-radius: 16px;

}

.button:hover {

  box-shadow: 0 12px 16px 0 rgba(0,0,0,0.24), 0 17px 50px 0 rgba(0,0,0,0.19);

}

form {border: 3px solid #f1f1f1; margin-left:400px;margin-right:400px;}

input[type=text], input[type=password] {

  width: 100%;

  padding: 12px 20px;

  display: inline-block;

  margin-bottom:18px;
```

```
  border: 1px solid #ccc;
  box-sizing: border-box;
}
button {
  background-color: #091425;
  color: white;
  padding: 14px 20px;
  margin-bottom:10px;
  border: none;
  cursor: pointer;
  width: 17%;
  border-radius:4px;
font-family:Montserrat;
}
button:hover {
  opacity: 0.8;
}
.cancelbtn {
  width: auto;
  padding: 10px 18px;
  background-color: #f44336;
}
.imgcontainer {
  text-align: center;
  margin: 24px 0 12px 0;
```

```
}
img.avatar {
  width: 30%;
  border-radius: 50%;
}
.container {
  padding: 16px;
}
span.psw {
  float: right;
  padding-top: 16px;
}
@media screen and (max-width: 300px) {
  span.psw {
     display: block;
     float: none;
  }
  .cancelbtn {
     width: 100%;
  }
}
.home{
margin:80px;
 width: 84%;
  height: 500px;
```

```
  padding-top:10px;
  padding-left: 30px;
}
.login{
        margin:80px;
        box-sizing: content-box;
  width: 84%;
  height: 420px;
  padding: 30px;
  border: 10px solid blue;
}
.left,.right{
 box-sizing: content-box;
 height: 400px;
 margin:20px;
 border: 10px solid blue;
}
.mySlides {display: none;}
img {vertical-align: middle;}
.slideshow-container {
max-width: 1000px;
  position: relative;
  margin: auto;
}
.text {
```

```
  color: #f2f2f2;
  font-size: 15px;
  padding: 8px 12px;
  position: absolute;
  bottom: 8px;
  width: 100%;
  text-align: center;
}
.dot {
  height: 15px;
  width: 15px;
  margin: 0 2px;
  background-color: #bbb;
  border-radius: 50%;
  display: inline-block;
  transition: background-color 0.6s ease;
}
.active {
background-color: #F8FD8F;
}
.fade {
  -webkit-animation-name: fade;
  -webkit-animation-duration: 1.5s;
  animation-name: fade;
  animation-duration: 1.5s;
```

```
}
@-webkit-keyframes fade {
  from {opacity: .4}
  to {opacity: 1}
}
@keyframes fade {
  from {opacity: .4}
  to {opacity: 1}
}
@media only screen and (max-width: 300px) {
  .text {font-size: 11px}
}
@importurl(https://fonts.googleapis.com/css?family=Anonymous+Pro);
html{
  min-height: 100%;
  overflow: hidden;
}
body{
  height: calc(100vh - 8em);
  padding: 4em;
color: rgba(255,255,255,.75);
  font-family: 'Anonymous Pro', monospace;
  background-color: black;
}
.line-1{
```

```
    position: relative;
    top: 50%;
    width: 200em;
    margin: 0 auto;
    border-right: 2px solid rgba(255,255,255,.75);
    font-size: 180%;
    text-align: center;
white-space: nowrap;
    overflow: hidden;
    transform: translateY(-50%);
}
.anim-typewriter{
  animation: typewriter 20s steps(200) 1s 1 normal both,
blinkTextCursor 500ms steps(200) infinite normal;
}
@keyframes typewriter{
  from{width: 0;}
  to{width: 200em;}
  z-index: -1;
}
@keyframesblinkTextCursor{
  from{border-right-color: rgba(255,255,255,.75);}
  to{border-right-color: transparent;}
}
</style>
```

<body>

 Let us build a UI where we get future predictions of Sales

Sales Forecast

<div class="topnav-right"style="padding-top:0.5%;">

app.py

```
import numpy as np

from flask import Flask,render_template,request

from tensorflow.keras.models import load_model #we are loading our model from keras

import pickle

app = Flask(__name__) #our flask app

scaler=pickle.load(open("Scaler_forecast.pkl","rb"))

model = load_model('sales_forecast.h5') #loading the model in the flask app

@app.route('/') #rendering html template

def home():

    return render_template("home.html") #rendering html template

@app.route('/intro')

def home1() :

    return render_template("intro.html") #rendering html template

@app.route('/predict')

def home2() :

    return render_template("web.html") #rendering html template

@app.route('/login',methods = ['POST']) #route for our prediction

def login() :
```

```
x_input=str(request.form['year']) #requesting the file
x_input=x_input.split(',')
    print(x_input)
    for i in range(0, len(x_input)):
x_input[i] = float(x_input[i])
    print(x_input)
x_input=np.array(x_input).reshape(1,-1)
temp_input=list(x_input)
temp_input=temp_input[0].tolist()
lst_output=[]
n_steps=10
i=0
    while(i<10):
            if(len(temp_input)>=10):
            #print("temp_input",temp_input)
x_input=np.array(temp_input[0:])
            print("{} day input {}".format(i,x_input))
x_input=np.expand_dims(x_input, axis=0)
x_input=scaler.transform(x_input)
            #x_input=x_input.reshape(1,-1)
x_input = x_input.reshape((1, n_steps, 1))
            #print("x_input.....",x_input)
yhat = model.predict(x_input, verbose=0)
yhat=scaler.inverse_transform(yhat)
            print("{} day output {}".format(i,yhat))
```

```
temp_input.extend(yhat[0].tolist())

temp_input=temp_input[1:]

        #print(temp_input)

lst_output.extend(yhat.tolist())

i=i+1

    else:

        print("Please give 10 number of inputs")

return render_template("web.html",showcase = 'The next day predicted value
is:'+str(lst_output[0][0]))

if __name__ == '__main__' :

app.run(debug = False,port=5000)
```