TRI-NIT Hackathon

Team MaLAI dama
Sriram S M
Shrung D N

Video Demo:
https://drive.google.com/file/d/15FVn6SbqZ_1Ih8tUFC8WV0zvJOUygt-v/view?usp=sharing

Problem statement

- Given the daily data of open price, close price, high, low and volume of IBM share price over the last 20 years, predict the close price of the next day.

Indicators used in Stock Trading

- Three types: Volatility, Trend, Momentum indicators
- These indicators use data such as open price, close price, volume of previous ‘x’ days and try to predict the same for the next few days.
- Tested many indicators such as:
  - Volatility - Bollinger Bands
  - Trend - Exponential Moving Average, Stochastic Moving Average
  - Momentum - Relative Strength Index, On Balance Volume

Exponential Moving Average

$$EMA = \text{Price}(t) \times k + EMA(y) \times (1 - k)$$

where:

$t = \text{today}$

$y = \text{yesterday}$

$N = \text{number of days in EMA}$

$k = 2 \div (N + 1)$

This is a weighted average of the close prices over the past N days, giving a higher weight to more recent days.

Relative Strength Index

$$RSI_{\text{step one}} = 100 - \left[\frac{100}{1 + \frac{\text{Average gain}}{\text{Average loss}}} \right]$$

Average gain is the average of %gain of all those days when %gain was positive
Average loss is the average of %loss of all those days when %loss was positive

On Balance Volume

The Formula For OBV Is

$$\text{OBV} = \text{OBV}_{prev} + \begin{cases} \text{volume}, & \text{if close} > \text{close}_{prev} \\ 0, & \text{if close} = \text{close}_{prev} \\ -\text{volume}, & \text{if close} < \text{close}_{prev} \end{cases}$$

where:

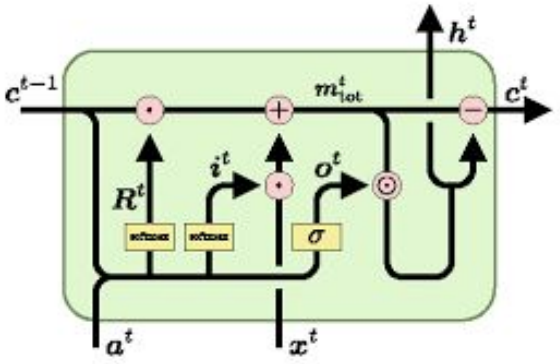
OBV = Current on-balance volume level

OBV_{prev} = Previous on-balance volume level

volume = Latest trading volume amount

Neural Network Model

- Tensorflow LSTM model
- LSTM - Long Short Term Memory
- Used to solve Time-Series problems
- The cell considers current input along with relevant data from previous inputs, thus making accurate predictions.



Structure of an LSTM cell

Our Model

```
model.summary()
```

```
Model: "sequential_3"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 lstm_3 (LSTM)               (None, 50)                10800

 dense_3 (Dense)             (None, 1)                 51

=================================================================
Total params: 10,851
Trainable params: 10,851
Non-trainable params: 0
_________________________________________________________________
```

Input Structure and Output of the model

Input:

- Normalized value of the day's Close Price (Between 0-1)
- Normalized value of the day's RSI (Between 0-1)
- Normalized value of the day's OBV (Between 0-1)
- These values were engineered in Google sheets using direct formulas as mentioned previously

Output:

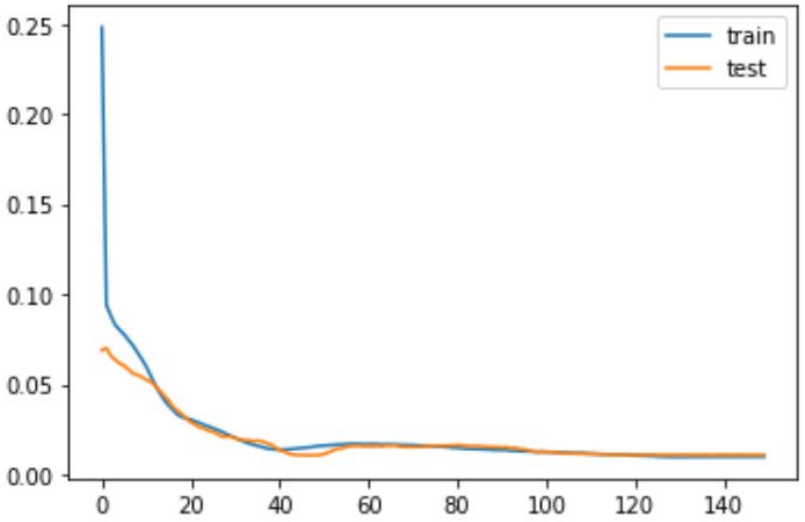
- Predicts the normalized value of the next day's Close Price (Between 0-1)

Results

Evaluation metrics over the test data (actual and predicted close price) :

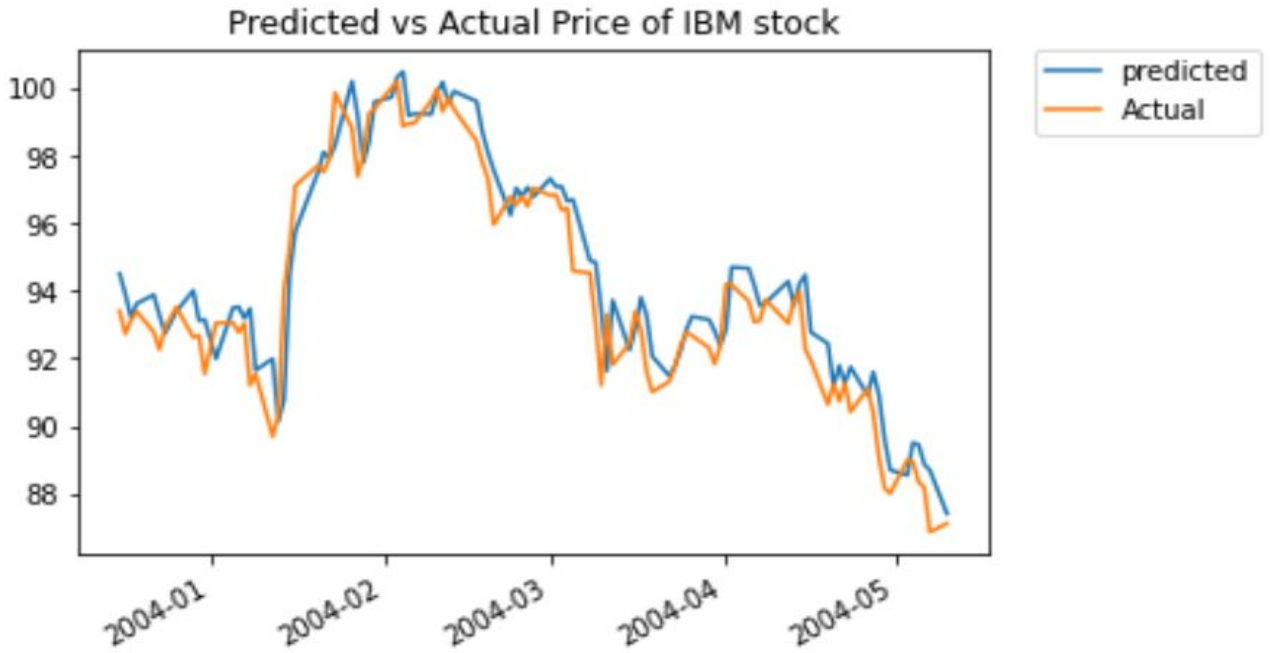
Test RMSE: 2.410

Test MAE: 1.743



MAE loss of predicted normalized close price value over 150 iterations of training

Results



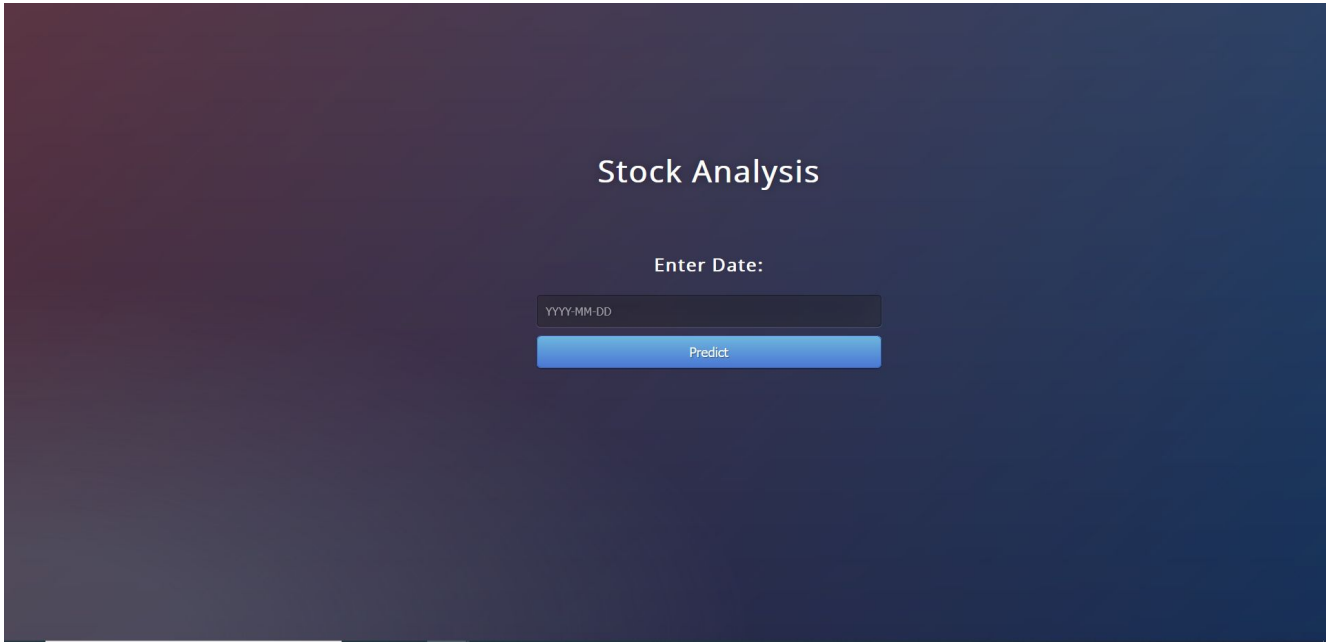
Predicted and actual stock price over 5 months

User Interface

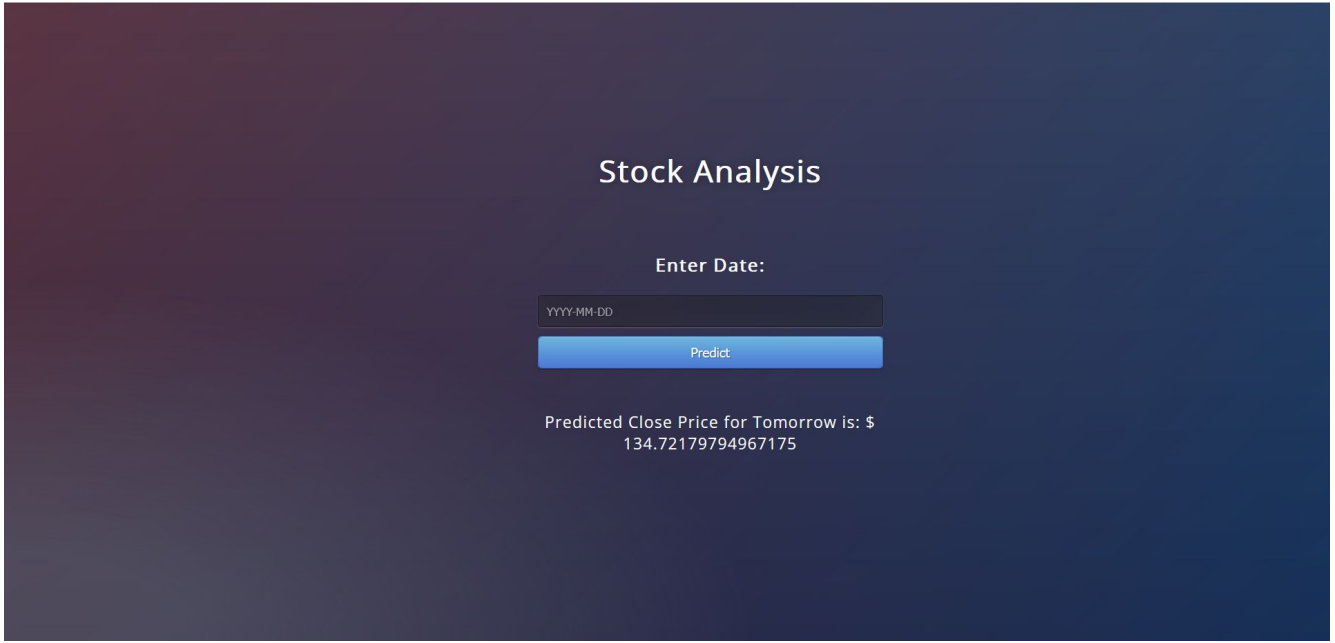
The UI consists of 3 web-pages (written in HTML which is integrated with the python code of model using Flask):

- Main webpage that the user sees on launch
- Web-page which shows the prediction if a correct date is input
- Web-page which shows an error if incorrect date is input

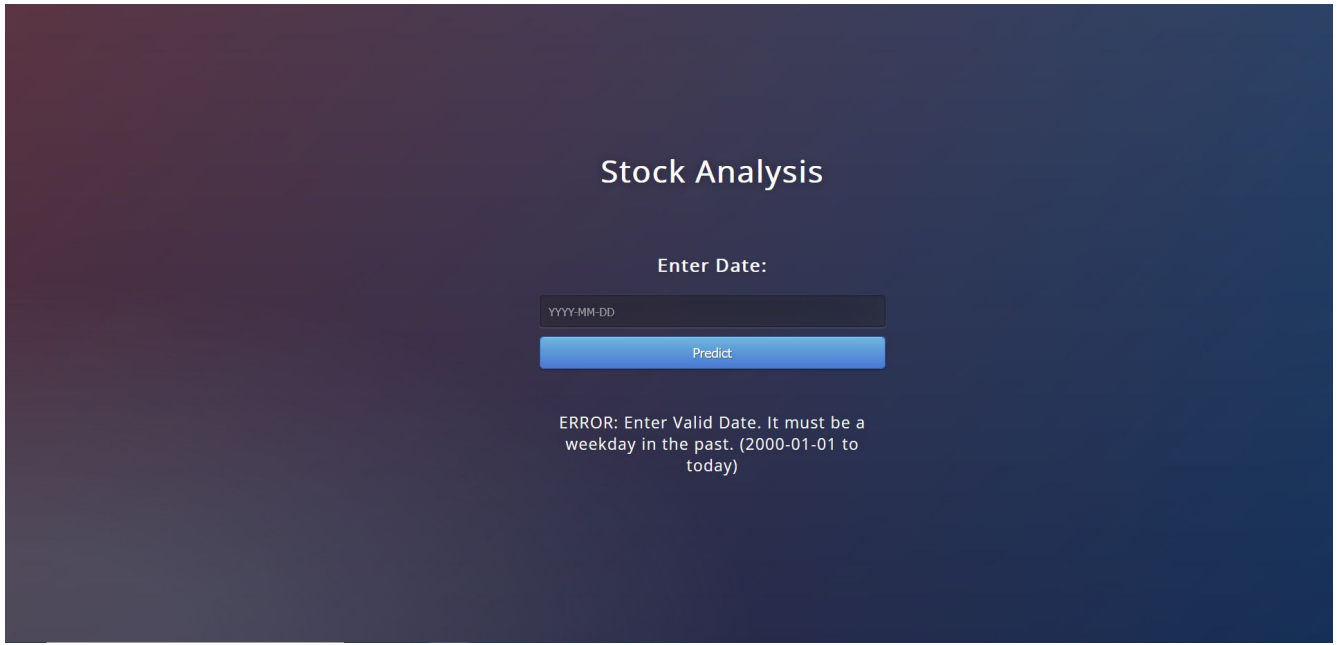
Main Web-page



Prediction Web-page



Error Web-page



Thank You