A PROJECT SYNOPSIS

on

STOCK PRICE PREDICTION

Submitted By

1) VRUTTI GOREGAONKAR (20)

2) PURVA PAWAR (49)

3) SAYALI SHINDE (55)

4) JATIN TIWARI (63)

Under the Guidance of

Dr. Divya Tamma

Department of Information Technology

Learn Live Achieve and Contribute

Saraswati Education Society's

SARASWATI COLLEGE OF ENGINEERING

Kharghar,Navi Mumbai

(Affiliated to University of Mumbai)

Academic Year :-2021-22

Saraswati College of Engineering, Kharghar

Vision:

To become center of excellence in Engineering education and research.

Mission:

To educate students to become quality technocrats for taking up challenges in all facets of life.

Department of Information Technology

Vision:

To create technically qualified talent through research to take up challenges in industries.

Mission:

1. To impart quality education.

2. To develop technical and managerial skills through training and modern teaching-learning process.

SARASWATI Education Society's

SARASWATI College of Engineering

Learn Live Achieve and Contribute

Kharghar, Navi Mumbai - 410 210.

(Approved by AICTE, reg. By MaharashtraGovt. DTE,Affiliated to Mumbai University)

PLOT NO. 46/46A, SECTOR NO 5, BEHIND MSEB SUBSTATION, KHARGHAR,NAVI MUMBAI-410210

Tel. : 022-27743706 to 11 * Fax : 022-27743712 * Website: www.sce.edu.in

CERTIFICATE

This is to certify that the requirements for the synopsis entitled""

Have been successfully completed by the following students:

Roll Numbers	*Name*
20	**VRUTTI GOREGAONKAR**
49	**PURVA PAWAR**
55	**SAYALI SHINDE**
63	**JATIN TIWARI**

In partial fulfillment of Sem –VI **Bachelor of Engineering of Mumbai University in Information Technology** ofSaraswati college of Engineering,Kharghar during the academic year 2021-22.

Internal Guide **External Examiner**

Dr. Divya Tamma

Project coordinator **Head of Department**

Prof. Ragini Sharma Prof. Maahi Khemchandani

Acknowledgement

A project is something that could not have been materialized without cooperation of many people. This project shall be incomplete if I do not convey my heartfelt gratitude to those people from whom I have got considerable support and encouragement.

It is a matter of great pleasure for us to have a respected **Dr. Divya Tamma** as my project guide.We are thankful to her for being constant source of inspiration.

We would also like to give our sincere thanks to **Prof. Maahi Khemchandani, H.O.D, Information Technology**Department**, Prof.Ragini Sharma**, **Project co-ordinator** for their kind support.

We would like to express our deepest gratitude to **Dr. Manjusha Deshmukh,**our principal of Saraswati college of Engineering, Kharghar, Navi Mumbai

Last but not the least I would also like to thank all the staffs of Saraswati college of Engineering (Information Technology Department) for their valuable guidance with their interest and valuable suggestions brightened us.

1) VRUTTI GOREGAONKAR	**(20)**
2) PURVA PAWAR	**(49)**
3) SAYALI SHINDE	**(55)**
4) JATIN TIWARI	**(63)**

STOCK PRICE PREDICTION

ABSTRACT

In this project we attempt to implement machine learning approach to predict stock prices. Machine learning is effectively implemented in forecasting stock prices. The objective is to predict the stock prices in order to make more informed and accurate investment decisions. We propose a stock price prediction system that integrates mathematical functions, machine learning, and other external factors for the purpose of achieving better stock prediction accuracy and issuing profitable trades.

There are two types of stocks. You may know of intraday trading by the commonly used term "day trading." Interday traders hold securities positions from at least one day to the next and often for several days to weeks or months. LSTMs are very powerful in sequence prediction problems because they're able to store past information. This is important in our case because the previous price of a stock is crucial in predicting its future price. While predicting the actual price of a stock is an uphill climb, we can build a model that will predict whether the price will go up or down.

In the finance world stock trading is one of the most important activities. Stock market prediction is an act of trying to determine the future value of a stock other financial instrument traded on a financial exchange.

INDEX

1.INTRODUCTION

This project is about Stock Price Prediction using machine learning. It helps you discover the future value of company stock and other financial assets traded on an exchange. The entire idea of predicting stock prices is to gain significant profits. Predicting how the stock market will perform is a hard task to do. There are other factors involved in the prediction, such as physical and psychological factors, rational and irrational behavior, and so on. All these factors combine to make share prices dynamic and volatile. This makes it very difficult to predict stock prices with high accuracy.

The financial market is a dynamic and composite system where people can buy and sell currencies, stocks, equities and derivatives over virtual platforms supported by brokers. The stock market allows investors to own shares of public companies through trading either by exchange or over the counter markets. This market has given investors the chance of gaining money and having a prosperous life through investing small initial amounts of money, low risk compared to the risk of opening new business or the need of high salary career.

The most common algorithms now are based on Recurrent Neural Networks (RNN), as well as its special type - Long-short Term Memory (LSTM) and Gated Recurrent Unit (GRU). Stock market is a typical area that presents time-series data and many researchers study on it and proposed various models. In this project, LSTM model is used to predict the stock price.

2.PROBLEM STATEMENT

The stock market prediction task is interesting as well as divides researches and academics into two groups those who believe that we can devise machanisms to predict the market and those who believe that the market is efficient and whenever new information comes up the market absorbs it by correcting itself, thus there is no space for prediction.

The stock market appears in the news in every day. The rate of investment and business opportunities in the stock market can increase if an efficient algorithm could be devised to predict the shot term price of an individual work.

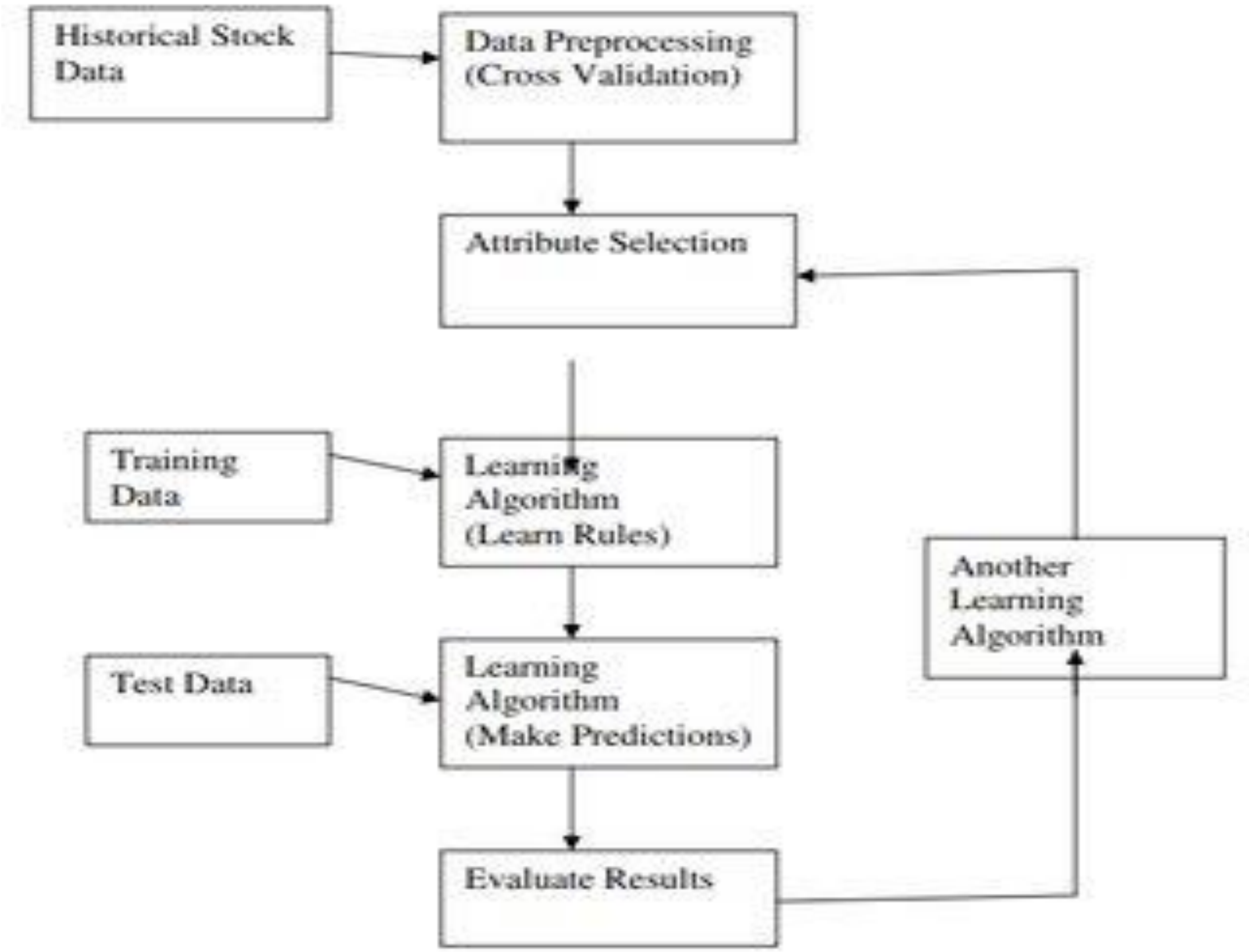
Time Series forecasting & modelling plays an important role in data analysis. Time series analysis is a specialized branch of statistics used extensively in fields such as Econometrics & Operation Research. Time Series is being widely used in analytics & data science. Stock prices are volatile in nature and price depends on various factors. The main aim of this project is to predict stock prices using Long short term memory (LSTM).

3. OBJECTIVE

The main objective of the our project is to develop robust framework for predicting stock price based on stock price data at 5 minute interval of time from the National Stock Exchange(NSE). So that they can have easy,to predict the stock prices in order to make more informed and accurate investment decisions.

A stock market prediction is described as an action of attempting to classify the future value of the company stock or other financial investment traded on the stock exchange. The forthcoming price of a stock of the successful estimation is called the Yield significant profit. This helps you to invest wisely for making good profits.

4. PROPOSED SYSTEM


Historical Stock Data
Data Preprocessing (Cross Validation)
Attribute Selection
Training Data
Learning Algorithm (Learn Rules)
Another Learning Algorithm
Test Data
Learning Algorithm (Make Predictions)
Evaluate Results

Block Diagram For Software

As per prediction system, developed in the stock price prediction to help investors in making financial decisions. In most researches it focuses on "lowest price buy", "highest selling price". On the "lowest buy" and "highest selling" strategy of stocks occurs when stocks are at the lowest price and sell shares when prices are highest. For stock price predictions ANN technique is used with back propagation the dataset was pre-processed and tuned up for real analysis. Hence, our admin can upload stock price history i.e. open price, highest price, lowest price and close price of the day. Paper will also focus on data preprocessing. Secondly, after preprocessing the data, System reads stock price history and gives input to the Back propagation algorithm. In addition, the proposed paper examines the use of the prediction system in real-world settings and issues associated with the accuracy of the overall values given. The back propagation gives output as final predicted rate comes. The proposed system can get the output of prediction list of stock price and graph of prediction table like that user can view the final predicted result. The successful prediction of the stock will be a great asset for the stock market institutions and will provide real-life solutions to the problems that stock investors face. To test BPNN method, mean square error is used to predict result and data reality. The backward propagation of errors, of back propagation, is a common method of training artificial neural networks and used in conjunction with an optimization method such as gradient descent. The method repeats in two phase cycle, propagation and weight update. During back-propagation phase the output after forward pass is compared with the expected output which is then used to adjust link weights. The output of the proposed system is to predict the list of stock price. At the end of system the user can view the final result of predicted value of stock market.

5. IMPLEMENTATION

System configuration:-This project can run on commodity hardware. We ran entire project on an Intel I5 processor with 8 GB Ram, 2 GB Nvidia Graphic Processor. First part of the is training phase which takes 10-15 mins of time and the second part is testing part which only takes few seconds to make predictions and calculate accuracy.

Hardware Requirements: • RAM: 4 GB • Storage: 500 GB • CPU: 2 GHz or faster • Architecture: 32-bit or 64-bit 5.1.2

Software requirements • Python 3.8.

- Operating System: windows 7/8/10.
- VS Code

CODE SNIPPETS

```
from django.shortcuts import render, HttpResponse

from django.views.decorators.csrf import ensure_csrf_cookie

from django.views.decorators.csrf import csrf_protect

from django.utils.decorators import method_decorator

from django.views.decorators.csrf import csrf_exempt

import numpy as np


import warnings

warnings.filterwarnings("ignore")
```

```
import matplotlib.pyplot as plt

import pandas as pd

import json


from scipy.fftpack import sc_diff

# Create your views here.

from .utils import prediction_image



def index(request):

    return render(request, 'index.html')

    # return HttpResponse("THis is home")



def about(request):

    return render(request, 'about.html')



def contact(request):

    return render(request, 'contact.html')
```

```
def services(request):
    return render(request, 'services.html')



def login(request):
    return render(request, 'login.html')



def stock_info(request):
    global df, df_reverse, close
    try:

        df = data_code
    except:

        df = data_name



    #Reversing the datafram because it shows new one first
    df_reverse=df
    df_reverse=df_reverse.loc[::-1].reset_index(drop = True)
    closed_data=df_reverse['close']
    closed_data= np.array(closed_data.to_numpy()) #Converting into numpy array
    closed_data = list(closed_data) # converting to list to add comma between
    close = float("{:.2f}".format(df.iloc[0,4])) #Eliminates number after decimals
    Stock_graph()

    equity_data=[{"date": df.iloc[0,0], "open": df.iloc[0,1], "high":df.iloc[0,2],
               "low": df.iloc[0,3], "close":df.iloc[0,4],
               "volume": df.iloc[0,5], "closed_data" : closed_data }]



    return render(request, 'stock_info.html', {'equity_data' : equity_data})




def Stock_graph():



    plt.plot(df_reverse['close'],color="green", label="Stock Price")
    plt.title(f"{equity_name} Share Price")
    plt.xlabel("Time")
    plt.ylabel("Closing Price" )
    plt.savefig(r'D:\ML Project\stock_price\price_prediction\static\img\Graph_images\closed_graph.png',dpi=900,facecolor='beige', bbox_inches="tight",pad_inches=0.3, transparent=True)
    plt.close()


    plt.bar(df_reverse.index,df_reverse ['volume'],color="blue", label="Predicted Price")
    plt.title(f"{equity_name} Share Volume Trade")
    plt.xlabel("Time")
    plt.ylabel("Volume Traded")
    plt.savefig(r'D:\ML Project\stock_price\price_prediction\static\img\Graph_images\volume_graph.png',dpi=900,facecolor='beige', bbox_inches="tight",pad_inches=0.3, transparent=True)
    plt.close()



#This function is called to avoid time taken for machine learning algorithm
@csrf_exempt
def prediction_graph(request):
    global prediction_price


    name = request.POST.getlist('name[]') #This will give me data which came from stock info template


    prediction_price = prediction_image(name[0])
    return HttpResponse(prediction_price)
```

```
def prediction(request):

    print(value, equity_name, close)

    prediction_price1 = [{'prediction_price' : prediction_price,'comapny_code': value,
'comapny_name' : equity_name, 'close_price':close}]

    return render(request, 'prediction.html', {'prediction_price': prediction_price1})




# This line ignore cookies

# if we don't return anything then it will throw an error



@csrf_exempt

def stock_name(request):


    global name1, value, data_code, data_name, data_overview, equity_name

    if request.method == 'GET':

        return HttpResponse("THis is get method")

        pass

    elif request.method == 'POST':
```

```
# access you data by playing around with the request.POST object
name = request.POST.getlist('name[]')


try:


    dict_items = d1.items()  # This gets all items in dictionaries


    for key, value1 in dict_items:


        if key == name[0]:
            value = value1
            equity_name=value1


except:
    # This get all value asscoiated with key
    name1 = list(d1.keys())[list(d1.values()).index(name[0])]
    equity_name=name1


    pass


try:
```

```
            data_code =
pd.read_csv(f'https://www.alphavantage.co/query?function=TIME_SERIES_DAILY&s
ymbol={name[0]}.BSE&datatype=csv&apikey=7KYND1T0YGSM56O8')

            # print(data_code)


data_overview=pd.read_json('https://www.alphavantage.co/query?function=OVERVIE
W&symbol={name[0]}.BSE&apikey=7KYND1T0YGSM56O8')

            # print(data_overview)

            return HttpResponse(value)  # Sending company name



        except:



            # This get all value asscoiated with key

            name1 = list(d1.keys())[list(d1.values()).index(name[0])]



            data_name =
pd.read_csv(f'https://www.alphavantage.co/query?function=TIME_SERIES_DAILY&s
ymbol={name1}.BSE&datatype=csv&apikey=7KYND1T0YGSM56O8')

            # print(data_name)


data_overview=pd.read_json('https://www.alphavantage.co/query?function=OVERVIE
W&symbol={name[0]}.BSE&apikey=7KYND1T0YGSM56O8')




            return HttpResponse(name1)  # SEnding Company code

import pandas as pd
import matplotlib.pyplot as plt

import numpy as np

import numpy as np



#Machine learing modules

from sklearn.preprocessing import MinMaxScaler

from tensorflow.keras.models import Sequential

from tensorflow.keras.layers import Dense, Dropout, LSTM



def hii():


    return("Hii Jatin")


def prediction_image(name):

    if(True):


        print(name)

        #Load Data

        company = name
```

```
df = pd.read_csv(f'https://www.alphavantage.co/query?function=TIME_SERIES_weekly&symbol={name}.BSE&datatype=csv&apikey=FH07SZXZ2ONHNOOO')

test_data = df

data = df[::-1].reset_index(drop = True) #reversing the dataframe




test_data =test_data.iloc[:80] #Selecting 80 data to predict

test_data=test_data[::-1]


#Prepare Data

scaler=MinMaxScaler(feature_range=(0,1)) #Converting in ragne of  and 1 so that values could't make bigger

scaled_data=scaler.fit_transform(data['close'].values.reshape(-1,1))


prediction_days = 80


x_train=[]

y_train=[]


for x in range(prediction_days, len(scaled_data)):

    x_train.append(scaled_data[x-prediction_days:x, 0])

    y_train.append(scaled_data[x, 0])
```

```
        x_train, y_train = np.array(x_train), np.array(y_train)

        x_train = np.reshape(x_train, (x_train.shape[0], x_train.shape[1], 1))


        #Build the Model

        model = Sequential()


        model.add(LSTM(units=50, return_sequences=True,
input_shape=(x_train.shape[1], 1)))

        model.add(Dropout(0.2))

        model.add(LSTM(units=50, return_sequences=True))

        model.add(Dropout(0.2))

        model.add(LSTM(units=50))

        model.add(Dropout(0.2))

        model.add(Dense(units=1)) #prediction of the next closing value


        model.compile(optimizer='adam', loss='mean_squared_error')

        model.fit(x_train, y_train, epochs=25, batch_size=32)


        '''Test the model accuracy on existing data'''


        #Load Test Data
```

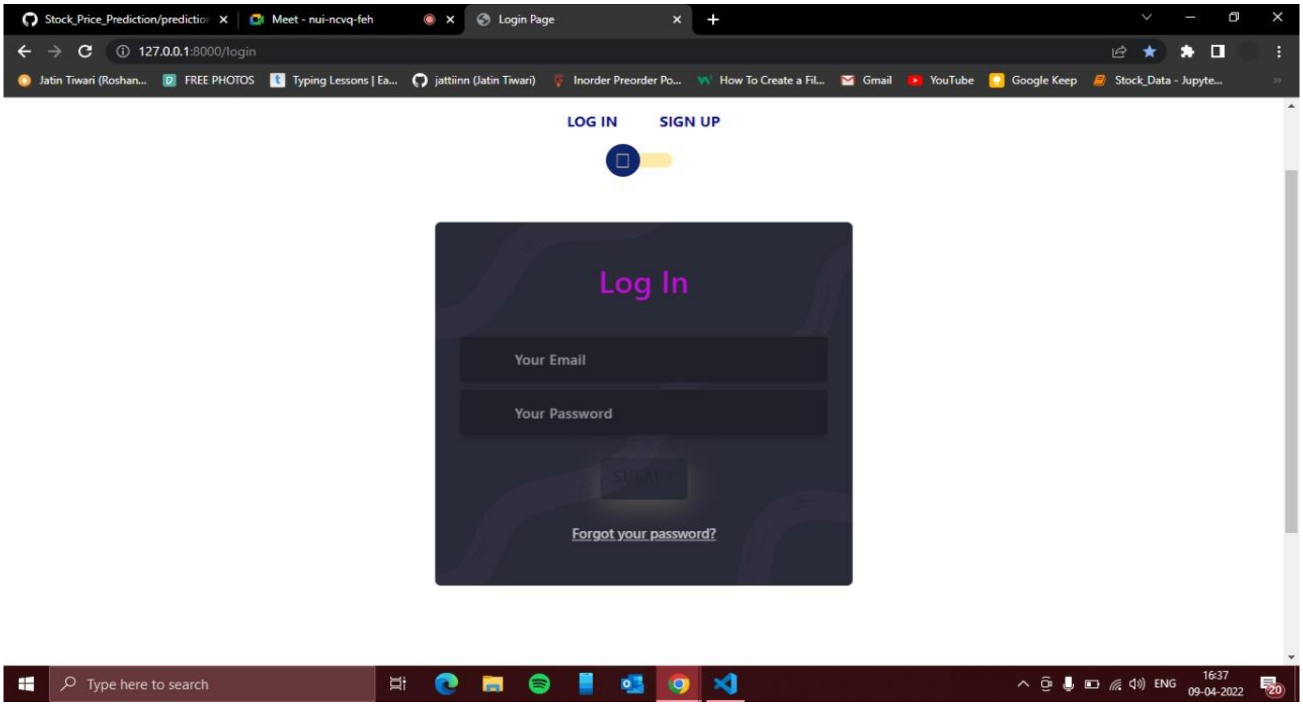
```
actual_prices=test_data['close'].values

total_dataset=pd.concat((data['close'], test_data['close']), axis=0)

model_inputs=total_dataset[len(total_dataset)-len(test_data)-prediction_days:].values
model_inputs = model_inputs.reshape(-1, 1)
model_inputs = scaler.transform(model_inputs)

# Make Predictions on Test Data
x_test=[]

for x in range(prediction_days, len(model_inputs)):
    x_test.append(model_inputs[x-prediction_days:x, 0])

x_test=np.array(x_test)
x_test=np.reshape(x_test, (x_test.shape[0], x_test.shape[1], 1))

predicted_prices=model.predict(x_test) #It is in in 0. something format
predicted_prices=scaler.inverse_transform(predicted_prices) #converting them into actual data
```

```
    # Plot the test predictions


    plt.plot(actual_prices, color = "black", label=f"Actual {company} Price")

    plt.title(f"{company} Share Price")

    plt.xlabel("Time")

    plt.ylabel(f"{company} Share Price")

    plt.legend()

    plt.savefig('D:\ML Project\stock_price\price_prediction\static\img\Graph_images\graph_actual.png',dpi=900,facecolor='beige', bbox_inches="tight",pad_inches=0.3, transparent=True)

    plt.close()



    plt.plot(predicted_prices, color="green", label=f"Predicted {company} Price")

    plt.title(f"{company} Predicted Share Price")

    plt.xlabel("Time")

    plt.ylabel(f"{company} Share Price")

    plt.legend()

    plt.savefig('D:\ML Project\stock_price\price_prediction\static\img\Graph_images\predicted_graph.png',dpi=900,facecolor='beige', bbox_inches="tight",pad_inches=0.3, transparent=True)

    plt.close()
```

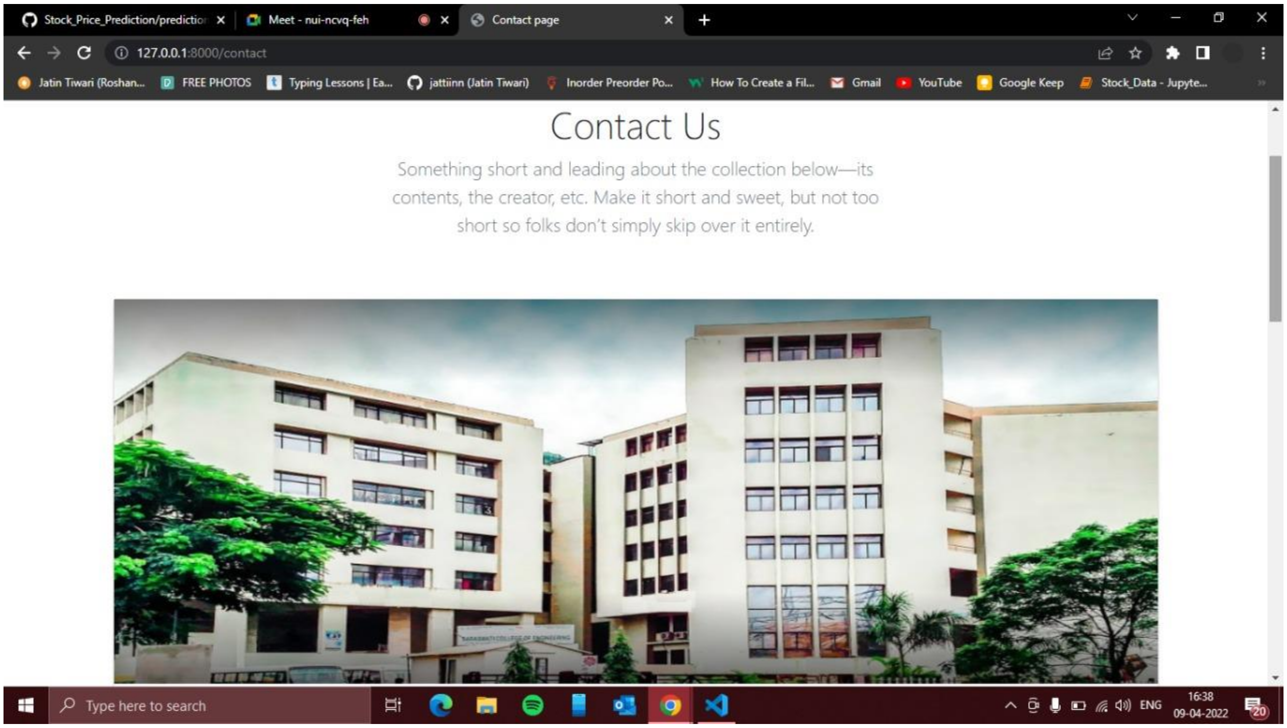
```
plt.plot(predicted_prices, color="green", label=f"Predicted {company} Price")
plt.plot(actual_prices, color="black", label=f"Actually {company} Price")


plt.title(f"{company} Predicted + Actual Share Price")
plt.xlabel("Time")
plt.ylabel(f"{company} Share Price")
plt.legend()
plt.savefig('D:\ML Project\stock_price\price_prediction\static\img\Graph_images\prediction_actual_graph.png',dpi=900,facecolor='beige', bbox_inches="tight",pad_inches=0.3, transparent=True)
plt.close()



#Predict Next Day
real_data = [model_inputs[len(model_inputs)-prediction_days:len(model_inputs+1), 0]]
real_data = np.array(real_data)
real_data=np.reshape(real_data, (real_data.shape[0], real_data.shape[1],1))
print(type(real_data))
prediction=model.predict(real_data)
prediction = scaler.inverse_transform(prediction)
prediction = prediction[0,0] #THis givs a single value
```

```
prediction=float("{:.2f}".format(prediction)) #it elminiates number after decimals
print(f"Prediction: {prediction}")
return prediction
```

6. OUTPUT SCREENSHOTS



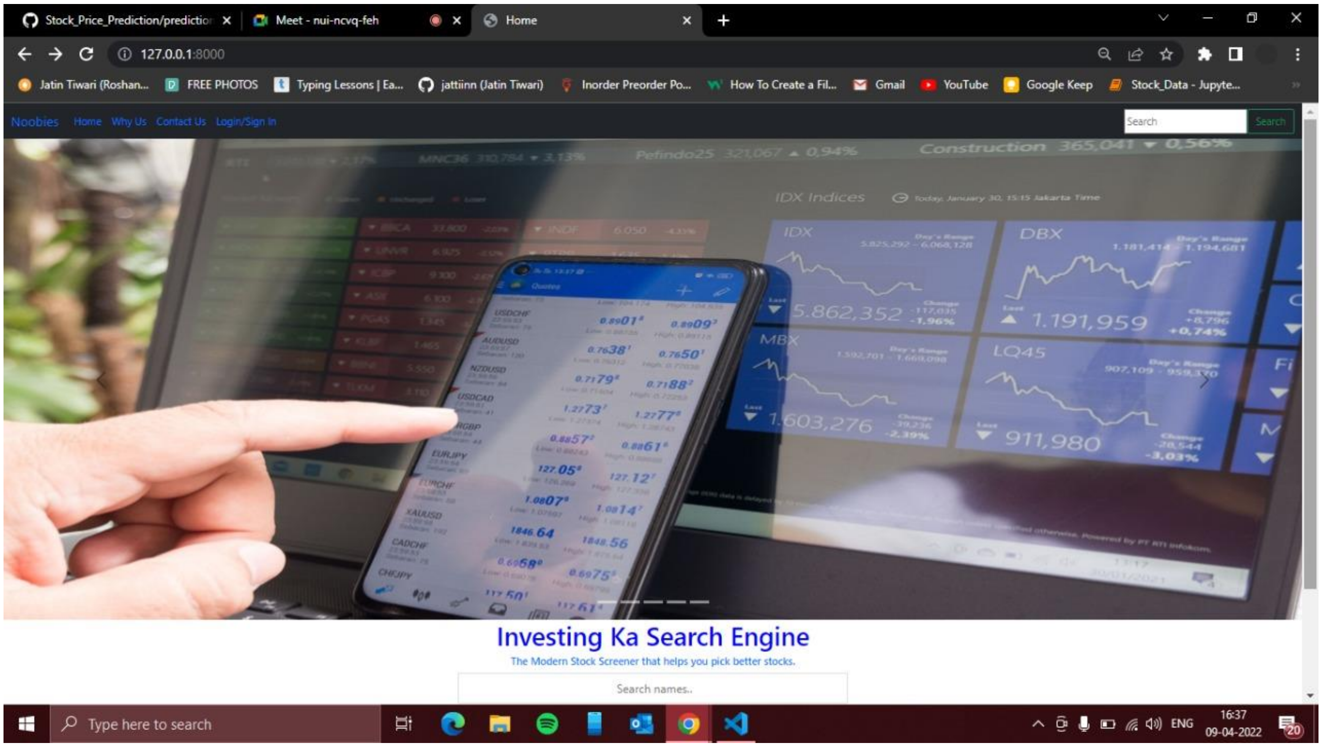
Login Page



Contact Us

Something short and leading about the collection below—its contents, the creator, etc. Make it short and sweet, but not too short so folks don't simply skip over it entirely.

Contact Page

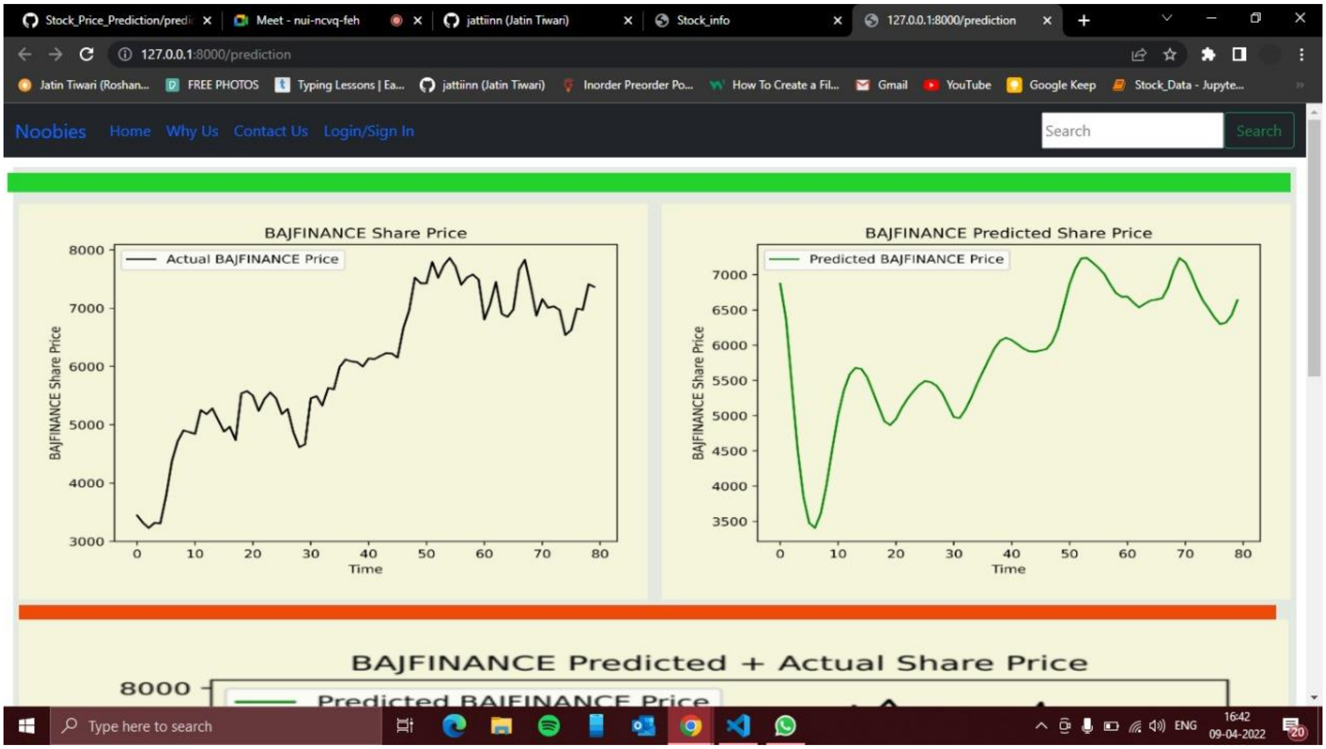


Investing Ka Search Engine

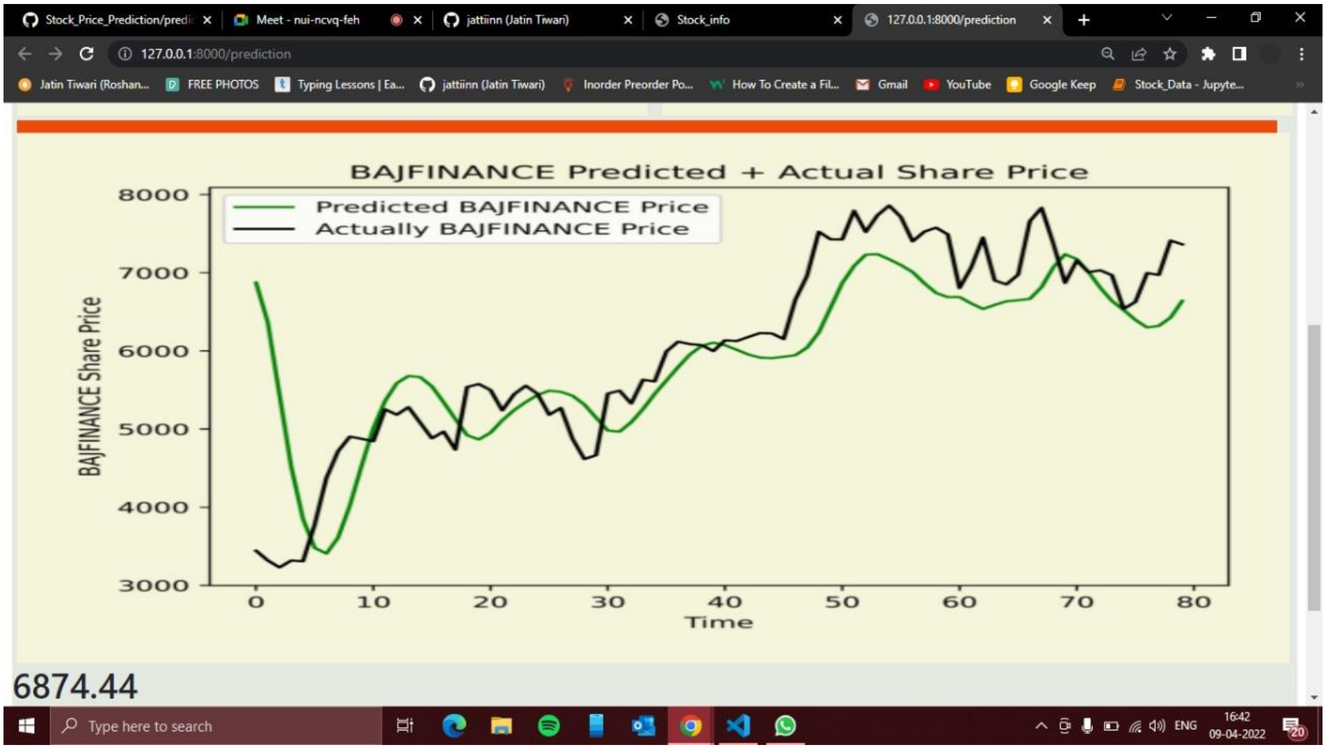
The Modern Stock Screener that helps you pick better stocks.

Search names..

Searching Page



Prediction Page


Stock_Price_Prediction/predi
Meet - nui-ncvq-feh
jattiinn (Jatin Tiwari)
Stock_info
127.0.0.1:8000/prediction
127.0.0.1:8000/prediction
Jatin Tiwari (Roshan...
FREE PHOTOS
Typing Lessons | Ea...
jattiinn (Jatin Tiwari)
Inorder Preorder Po...
How To Create a Fil...
Gmail
YouTube
Google Keep
Stock_Data - Jupyte...
BAJFINANCE Predicted + Actual Share Price
Predicted BAJFINANCE Price
Actually BAJFINANCE Price
BAJFINANCE Share Price
8000
7000
6000
5000
4000
3000
0
10
20
30
40
50
60
70
80
Time
Type here to search
ENG
16:42
09-04-2022
20

6874.44

7. RESULT

The sample were tested below. The Testing and training accuracies are tabulated below for epochs.

Reliance

Epoch	**Accuracy**
1	75.67
2	77.87
3	85.65

HDFC

Epoch	**Accuracy**
1	80.45
2	85.35
3	88.67

We have applied datasets belonging to Reliance and HDFC Stocks and achieved above 80% accuracy for these datasets.

8. CONCLUSION

In this project, we are predicting closing stock price of any given organization, we developed a web application for predicting close stock price using LMS and LSTM algorithms for prediction. We have applied datasets belonging to Google, Nifty50, TCS, Infosys and Reliance Stocks and achieved above 85% accuracy for these datasets.

Machine learning techniques such as fundamental analysis, technical and sentiment analysis for stock price prediction can be very successful when applied correctly. It is important to apply these models in a way that will reduce errors and maximize results. One approach you could take is utilizing data from past transactions in your market or combining it with other investment strategies like technical analysis, fundamental analysis, and quantitative easing.

The use of a model based on historical data provides more solid information because it has been previously tested for accuracy in the market; this eliminates incorrect or incomplete data which provides unreliable predictions about future prices movements. Machine learning techniques are also useful at analyzing how different stocks move together over time–for example, if there’s an upswing due to changing sentiments then investors would want to know what caused the positive change so they can invest with more confidence.

9. FUTURE SCOPE

Future scope of this project will involve adding more parameters and factors like the financial ratios, multiple instances, etc. The more the parameters are taken into account more will be the accuracy. The algorithms can also be applied for analyzing the contents of public comments and thus determine patterns/relationships between the customer and the corporate employee.

The use of traditional algorithms and data mining techniques can also help predict the corporation performance structure as a whole. In the future, we plan to integrate neural network with some other techniques such as genetic algorithm or fuzzy logic.

Genetic algorithm can be used to identify optimal network architecture and training parameters. Fuzzy logic provides the ability to account for some uncertainty produced by the neural network predictions. Their uses in conjunction with neural network could provide an improvement for stock market prediction.

10. REFERENCES

[1] Stock Price Prediction Using LSTM on Indian Share Market by Achyut Ghosh, Soumik Bose1, Giridhar Maji, Narayan C. Debnath, Soumya Sen

[2] S. Selvin, R. Vinayakumar, E. A. Gopalkrishnan, V. K. Menon and K. P. Soman, "Stock price prediction using LSTM, RNN and CNN-sliding mmunications and Informatics, 2017.

[3] Murtaza Roondiwala, Harshal Patel, Shraddha Vwindow model," in International Conference on Advances in Computing, Coarma, "Predicting Stock Prices Using LSTM" in Undergraduate Engineering Students, Department of Information Technology, Mumbai University, 2015.

[4] Xiongwen Pang, Yanqiang Zhou, Pan Wang, Weiwei Lin, "An innovative neural network approach for stock market prediction", 2018

[5] Ishita Parmar, Navanshu Agarwal, Sheirsh Saxena, Ridam Arora, Shikhin Gupta, Himanshu Dhiman, Lokesh Chouhan Department of Computer Science and Engineering National Institute of Technology, Hamirpur – 177005, INDIA - Stock Market Prediction Using Machine Learning.

[6] Pranav Bhat Electronics and Telecommunication Department, Maharashtra Institute of Technology, Pune. Savitribai Phule Pune University - A Machine Learning Model for Stock Market Prediction.

[7] Anurag Sinha Department of computer science, Student, Amity University Jharkhand Ranchi, Jharkhand (India), 834001 - Stock Market Prediction Using Machine Learning.