Design Credit- Project work.

Project - Stock market Prediction using Deep learning.

Instructor- Dr. Gaurav Kumar.

Aim: Predicting the stock market in a different way from the usual techniques, i.e using sentiments from twitter.

Stock Market prediction.

A stock market prediction is an attempt to forecast the future value of an individual stock, a particular sector or the market, or the market as a whole. These forecasts generally use fundamental analysis of a company or economy, or technical analysis of charts, or a combination of the two. Here, we are going to perform forecasting based on the sentiments received from the twitter , and this project is just a illustration of what can be implemented on large scale for better results.

Task -1-

In the first task, we were required to look into the techniques that were previously used by different experiments in similar tasks. I looked into 10 projects similar to ours and listed down the data and information I got from them, by reading their research papers.

The link to the sheet is-
https://docs.google.com/spreadsheets/d/1qixhT3r1niXIPN7t4zYQZ-YYEpElu7JAjQNf401GU04/edit?usp=sharing
This sheet was having all the details and method of the implementation of different experiments they did and how they managed to get the datasets.
After this, we discussed the methods used in these papers.

Task -2-

Now after analyzing the papers and their experiments, we decided to look into the sentimental effects of social media, on the prizes of stocks, Like there was news prevailing about the prize hike of dogecoin due to the statement from the great Elon Musk. So we were planning to take this as our approach for predicting the stocks using the sentiments of the stock market.

https://digitalcommons.newhaven.edu/cgi/viewcontent.cgi?article=1329&context=americanbusinessreview

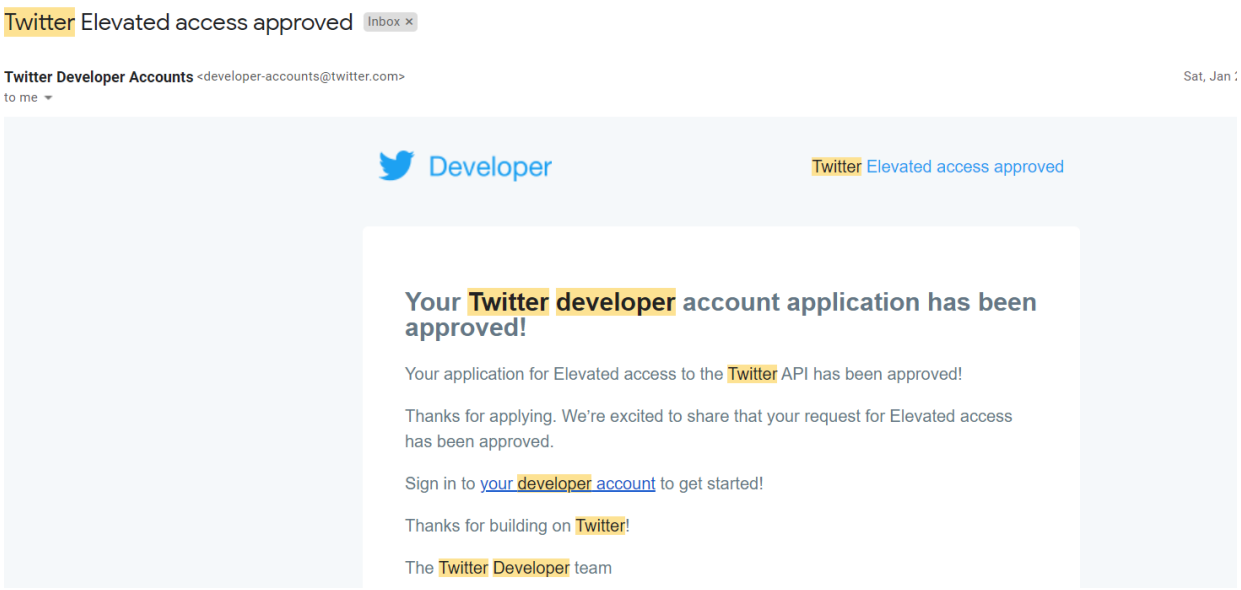
Task-3:

Now our major task is to find the dataset and thinking of approach for the extracting the sentiments out of it.

I tried using different types of resources available on the internet for the tweets extraction , but we needed tweets for a day and also based on particular keywords, i.e corona and stocks both together.

What we did then?

I read about the twitter developer access and hence applied for it and received it after telling them requirements and necessity , screenshot of which is attached-



Then I tried extracting the tweets through the use of this account ,

(Colab link for the file of extraction of tweets from developer account - Click here.)

```
consumer_key = "[redacted]"
consumer_secret = "[redacted]"
access_key= "[redacted]"
access_secret = "[redacted]"

auth = tw.OAuthHandler(consumer_key, consumer_secret)
auth.set_access_token(access_key, access_secret)
startDate = datetime.datetime(2022, 1, 31,  0, 0, 0)
endDate =   datetime.datetime(2022, 4,22, 0, 0, 0)
api = tw.API(auth,wait_on_rate_limit=True)

csvFile = open('tweets_mon_1.csv', 'a')
csvWriter = csv.writer(csvFile)

search_words = "#Corona" and "#Sensex"      # enter your words
new_search = search_words + " -filter:retweets"


for tweet in tw.Cursor(api.search,q=new_search,count=100,lang="en").items():
  if tweet.created_at < endDate and tweet.created_at > startDate:
    csvWriter.writerow([tweet.created_at, tweet.text.encode('utf-8'),tweet.user.screen_name.encode('utf-8'), tweet.user.location.encode('utf-8')])
```

but again there is problems from twitter side –

1. Twitter was not allowing to get tweets older than one week at a time for a particular time.
2. Twitter is giving the tweets for different type of accounts without considering the no. of followers of the tweet's writer.
3. Tweets are filled with lots of useless emojis and expressions and difficult for us to interpret the sentiments of the user.
4. The dataset is not possible for long periods of time.

What I did to sort these?

1. For the number of days it is not possible to do something without some very different access but without presence of any help from any where I , after consultation with my Sir, decided to create a prototype of the model instead of working model.
2. For different type of account we sorted the extracted tweets on the basis of number of followers of that particular account, then used that data for each day.
3. We used cleanup text module from python for cleaning up the tweets so we can

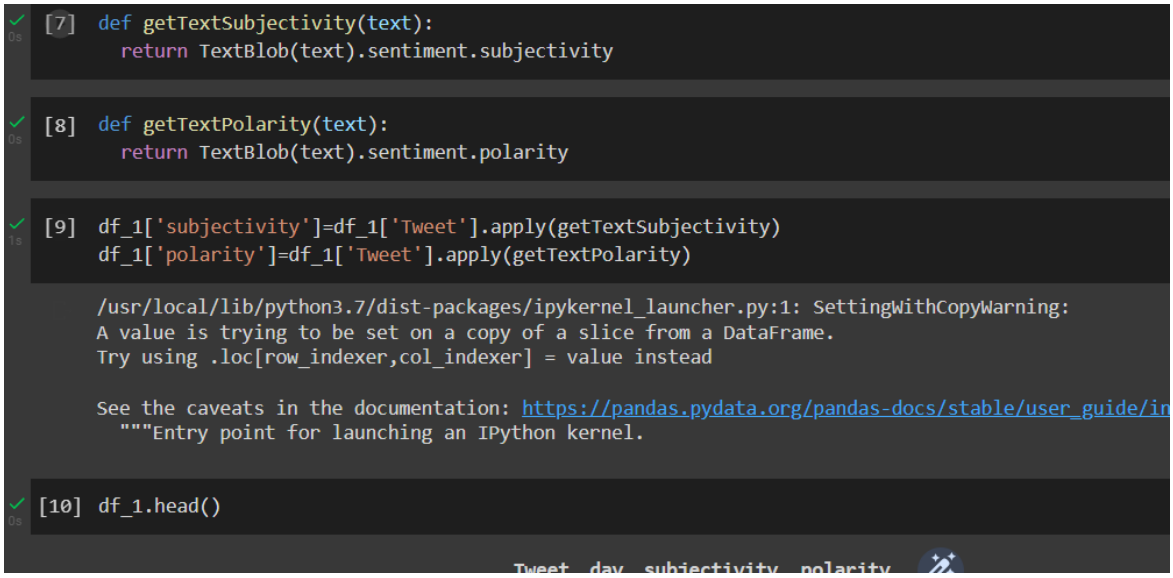
```
def cleanup(text):
    temp = text.lower()
    temp = re.sub("'", "", temp) # to avoid removing contractions in english
    temp = re.sub("@[A-Za-z0-9_]+","", temp)
    temp = re.sub("#[A-Za-z0-9_]+","", temp)
    temp = re.sub(r'http\S+', '', temp)
    temp = re.sub('[()!?]', ' ', temp)
    temp = re.sub('\[.*?\]',' ', temp)
    temp = re.sub("[^a-z0-9]"," ", temp)
    temp=re.sub(r'RT : ','',temp)
    return temp
```

get the exact sentiments from them.

4. The dataset problem is still unsolved and the model is just the working prototype of the project.

Sentiments extraction-

Now we have the tweets ready for extraction of sentiments from them , and for that we used python machine learning library and that applied on cleaned text.



```
[7]  def getTextSubjectivity(text):
       return TextBlob(text).sentiment.subjectivity

[8]  def getTextPolarity(text):
       return TextBlob(text).sentiment.polarity

[9]  df_1['subjectivity']=df_1['Tweet'].apply(getTextSubjectivity)
     df_1['polarity']=df_1['Tweet'].apply(getTextPolarity)

     /usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:1: SettingWithCopyWarning:
     A value is trying to be set on a copy of a slice from a DataFrame.
     Try using .loc[row_indexer,col_indexer] = value instead

     See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/in
       """Entry point for launching an IPython kernel.

[10] df_1.head()

                                                    Tweet  day  subjectivity  polarity
```

Hence we got the sentiments for the required tweets on our keywords, i.e corona and stocks both together.

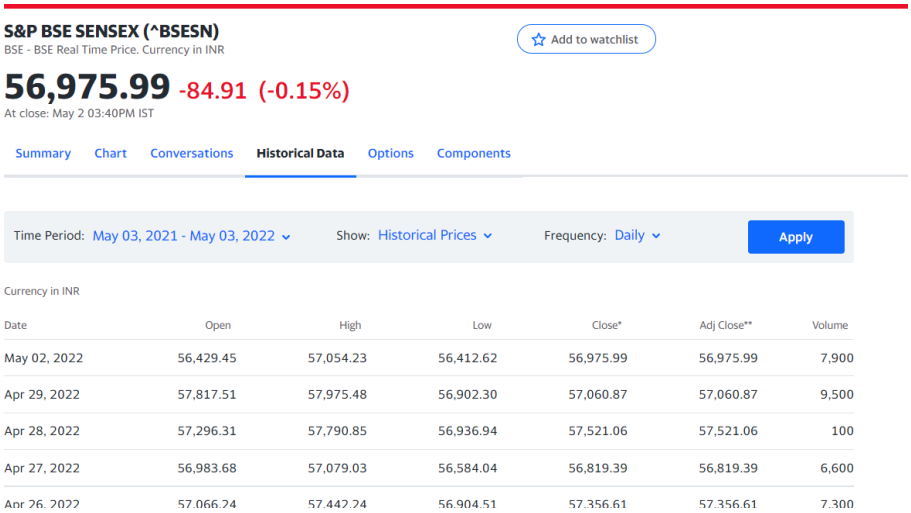
Task - 4

Now, our goal is now get the stock price and I had chosen sensex as our reference.

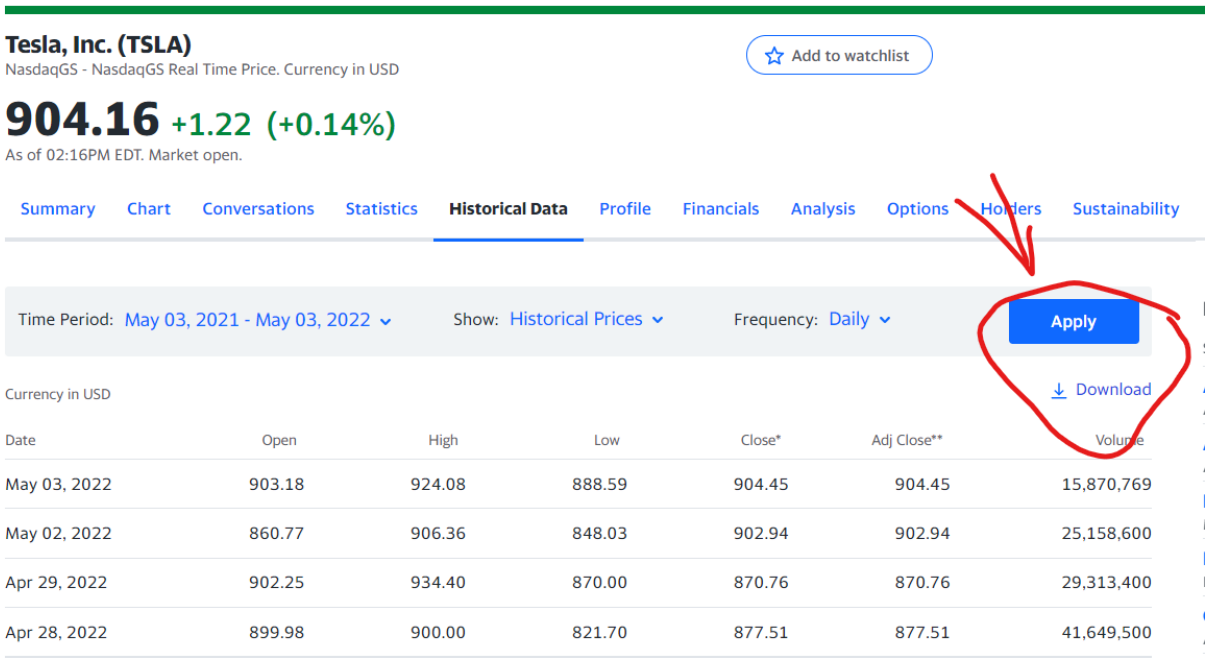
How we got the price?

We had used yahoo finance, but again we faced problem that, without specifying the reason yahoo is not allowing us to download the dataset for BSE SENSEX.

Illustration- For Sensex (No download option)



For others we had option to download-



So then we used inspecting technique for this website and then we copied that requesting url from tesla and applied it to the BSE SENSEX , to get the data.

Now after getting data like this -

	A	B	C	D	E	F	G
1	Date	Open	High	Low	Close	Adj Close	Volume
2	31-01-2022	57845.91	58257.63	57746.15	58014.17	58014.17	7400
3	01-02-2022	58672.86	59032.2	57737.66	58862.57	58862.57	14000
4	02-02-2022	59293.44	59618.51	59193.05	59558.33	59558.33	8500
5	03-02-2022	59528.16	59557.87	58653.94	58788.02	58788.02	7100
6	04-02-2022	58918.65	58943.62	58446.95	58644.82	58644.82	7300
7	07-02-2022	58549.67	58707.76	57299.05	57621.19	57621.19	11500
8	08-02-2022	57799.67	57925.82	57058.77	57808.58	57808.58	8900
9	09-02-2022	58163.01	58507.61	58105.18	58465.97	58465.97	5900
10	10-02-2022	58810.53	59060.24	58332.28	58926.03	58926.03	6700
11	11-02-2022	58447.15	58447.15	57914.1	58152.92	58152.92	5700
12	14-02-2022	56720.32	57191.91	56295.7	56405.84	56405.84	8300
13	15-02-2022	56731.56	58211.38	56438.47	58142.05	58142.05	10000
14	16-02-2022	58310.68	58569.22	57780.28	57996.68	57996.68	7300
15	17-02-2022	58217.69	58346	57635.43	57892.01	57892.01	6400
16	18-02-2022	57488.39	58175.35	57488.39	57832.97	57832.97	9800
17	21-02-2022	57551.65	58141.96	57167.02	57683.59	57683.59	6600

^BSESN (1)

We need to get some single value from this dataset to feed that into our models and predict or train our models accordingly.

So we defined a factor for that , i.e -

What is Factor here?

We calculated a single value for representing the stock price fluctuations which can consider both the opening and closing price of the market.

Our used formula -

```
Factor = (Closing Price - Opening Price*Volume/(High of the day - low of the day)
```

Then we created a column for each of the dataset and added this factor into it, and after that we standarized that value using python library for preprocessing the dataset , and then we used that factor for predicting through the models.

```
] from sklearn import preprocessing
  X_data= Stock_data['Factor']
  standard = preprocessing.scale(X_data)
  print(standard)
```

```
[-0.6225022  -0.6225022  -0.6225022   0.07509179  2.09929878  0.14145527
  0.84920479 -1.29754403]
```

Now we have both factor and the sentiments and that we used as the training data and training output, and then we splitted the dataset into training and testing data and the imported all the models and trained different models using that ,

Colab file for the whole source code-

[Click here]

We also downloaded the pickle file , in case we want to create some UI for our model.

So , that’s all for this project and would like to thank institute, department and my supervisor for this wonderful project.

Thank you.