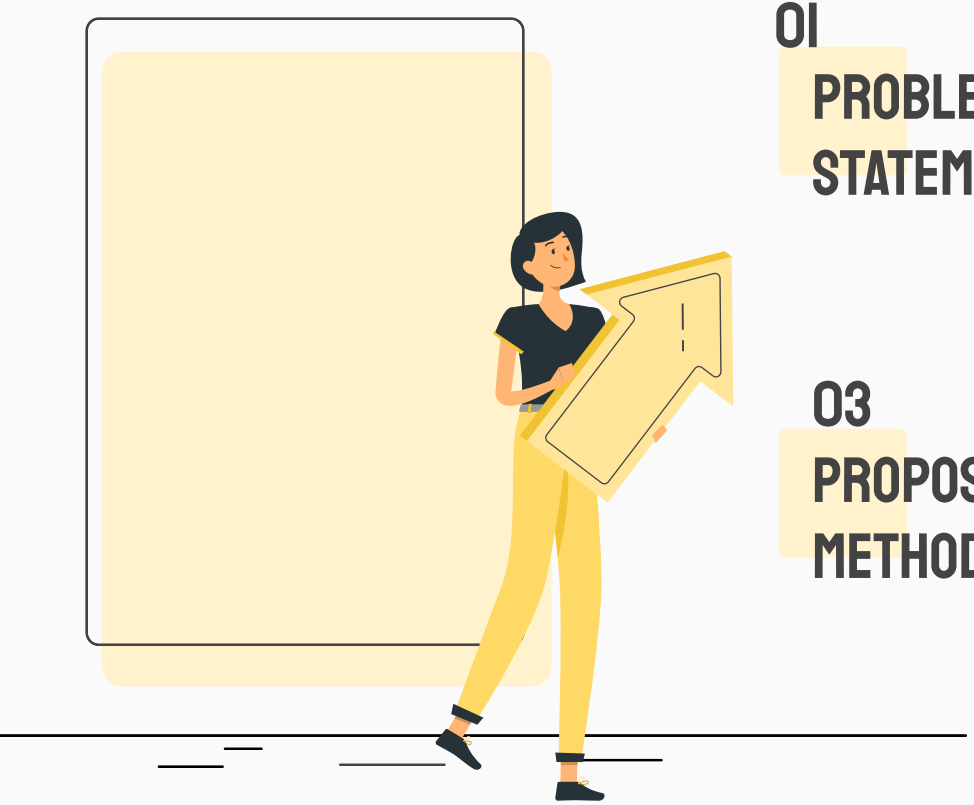
STOCK PRICES FORECASTING

Mohammed Faisal BT18GCS067
Somanath Vamshi BT18GCS076
Arjun Bakshi BT18GEC134



01
PROBLEM
STATEMENT

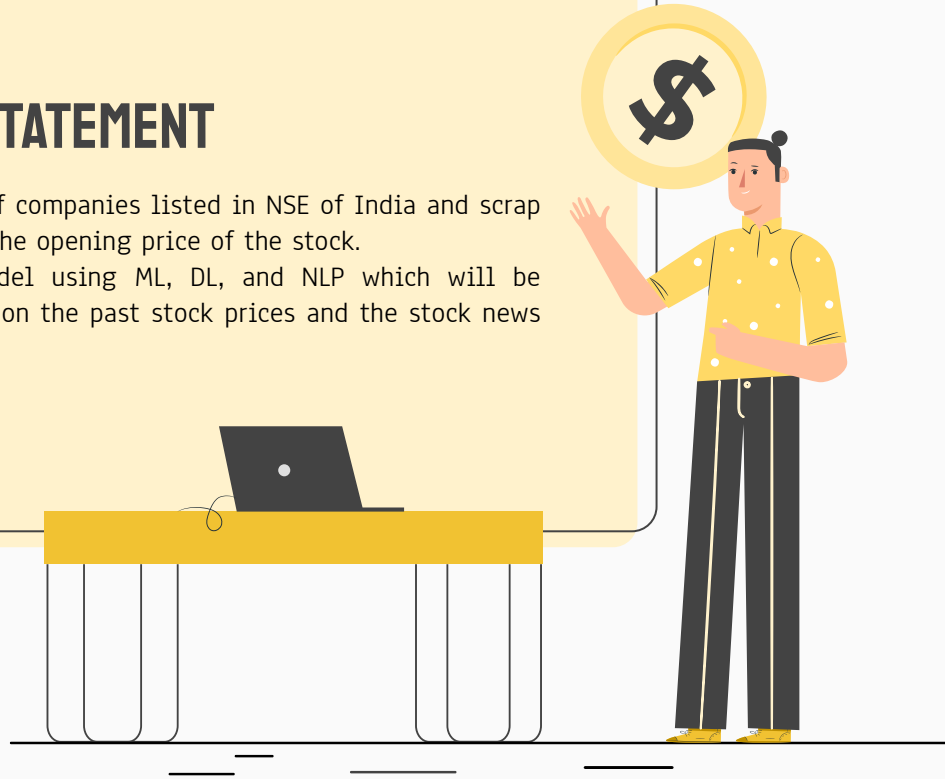
02
LITERATURE
REVIEWS

03
PROPOSED
METHODOLOGY

04
RESULTS

PROBLEM STATEMENT

The goal of our work is to collect stock prices of companies listed in NSE of India and scrap the news data related to that stock and predict the opening price of the stock.
In this paper we have proposed a hybrid model using ML, DL, and NLP which will be predicting the opening price of the stock based on the past stock prices and the stock news affecting that company.



OBJECTIVES

TECHNICAL ANALYSIS

This will include financial data like open value, close value, high, low, volume traded, turnover.
Data will be extracted from pandas library,preprocessed and then feed into the LSTM network to predict the opening price of the stock.

SENTIMENT ANALYSIS

This will include web scraping of news from google news or screener and evaluating the keywords using NLP tools to understand the sentiments of the market and calculate the sentiment score.
These scores will then be added into our dataset.

LITERATURE REVIEW

01
COMPARISON BETWEEN ML AND DL MODELS

- SVM vs ANNs (combined SVM with PSO optimization)
- ARIMA, SARIMA, ANN, CNN and RNN comparisons.[CNN,RNN better]

Findings: ML models are not very accurate for time series data

Sentiment analysis is not considered

02
COMPARISONS BETWEEN DIFFERENT DEEP LEARNING MODELS

- Variating lookback periods in RNNs
- Univariate and multivariate series

Findings: Accuracy of multivariate models lags behind univariate ones.

Sentiment analysis is again not considered

03
MODELS USING NLP DATA

- LSTM and GRU compared(has inspired our base model)
- Feature Engineering in addition to model.

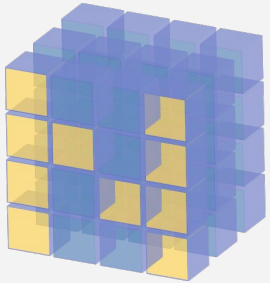
Findings: Stock news carries important data for predictions

Sentiment analysis proves to give better predictions

TECHNOLOGY

matplotlib

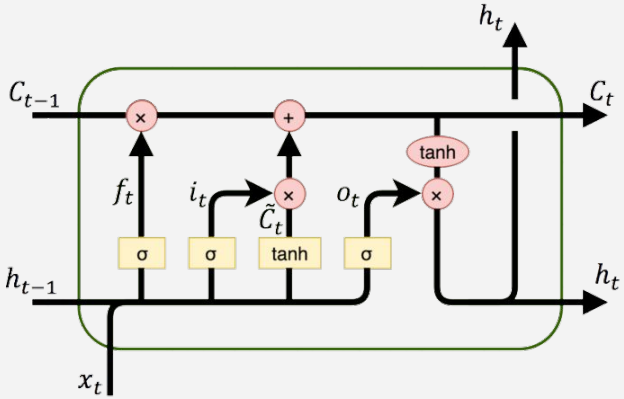
seaborn

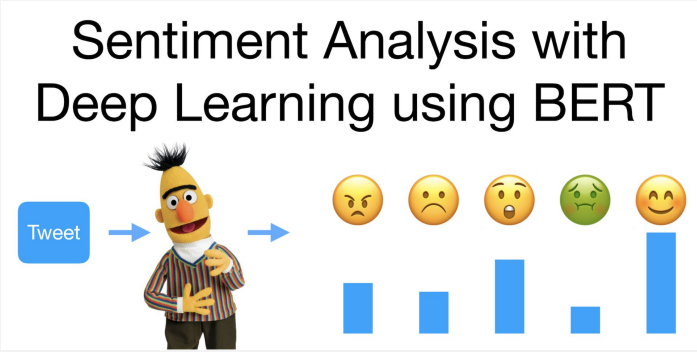


NumPy

TECHNOLOGY

Keras



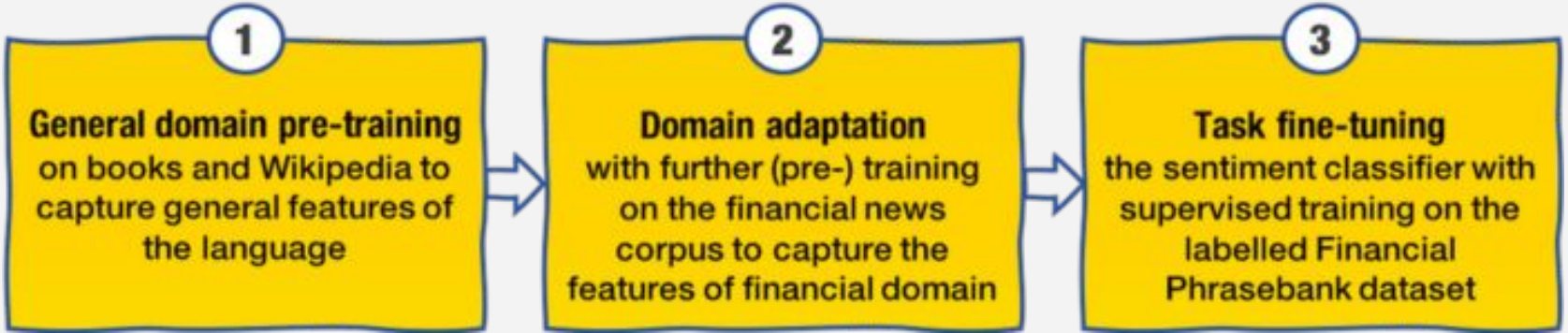


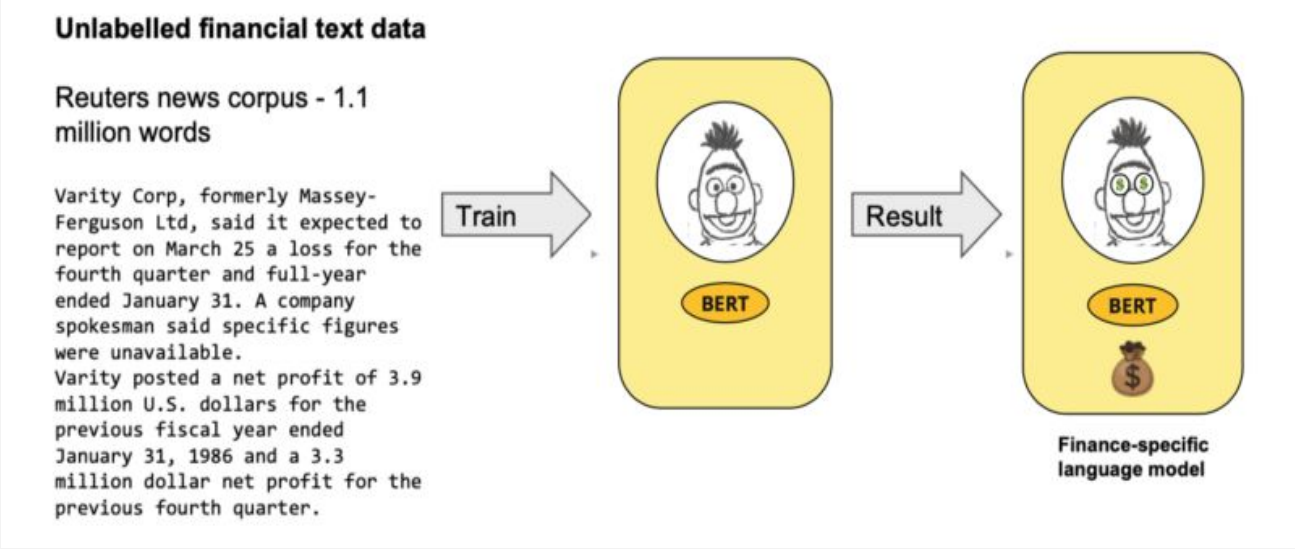
DATA EXTRACTION

- Collection of Data(Technical Analysis):
  - Historical data of the stock is collected from Yahoo Finance Website which consists of Open, High, Low, Volume Traded, Close price of the stock.
  - Data taken from 2012 August to 2021 April
- Collection of News Data:
  - News data is collected from Moneycontrol website for that particular day of the targeting stock.
  - Data is scraped using the beautiful soup library in python.
  - Data taken from 2012 August to 2021 April

FINBERT

- BERT was released recently (October 2018 by Google). FinBERT is BERT fine-tuned specifically for financial sentiment.
- This model was used by us primarily due to the failure of VADER to output scores that had correlation with the open prices.
- BERT overcomes the challenge posed by RNNs of relating words that are far apart.
- It is available pre-trained with a vocab of 30873 words, 12 hidden layers with 768 cells with an hidden activation of 'relu' and hidden dropout probability of 0.1.
- These are pre-determined for financial sentiments, we have not experimented with parameters or changed the model.





After getting the finbert model we have to give our customised headlines into this bert model then this model predicts the sentiment of the headline .
Like 0: for Neutral , 1: for Positive , -1 : negative

Example: there is a shortage of capital, and we need extra financing
FinBERT predicted sentiment: negative

DATA PREPROCESSING

Preparing the Dataset - Getting ready for NLP

1. After extracting the **data(**Date, open, close, high, low, volume) and headlines/news of the date range.
2. Compiling the headlines that were scraped on a particular date into one array.
3. All such arrays were appended to a bigger array which had the compiled news for each specific date w.r.t to the index within the bigger array.

```
2552
                                                HeadLine      day     Dates
19938  Sensex rises 200 pts; TCS, SBI, ICICI Bank, In...  Tuesday  8/7/2012
['RBS sees Nifty hitting 5,700 mark by December', 'Sensex ends 1% higher two-day in a row on reform h
opes', 'Expect Nifty to test 5350 soon', 'Sensex rises 200 pts; TCS, SBI, ICICI Bank, Infosys gain
2%']


                                            HeadLine     day      Dates
0  Mutual funds make fresh buy in these 16 stocks...  Friday  4/16/2021
['Mutual funds make fresh buy in these 16 stocks, exit 8 in March', 'Where are the ultra-rich investi
ng? Over 20 stocks from top 5 PMS funds outperformed Nifty in March', 'COVID takes toll, over 50% Nif
ty50 stocks are down 10-20% from highs', 'Trade Spotlight: What should investors do with Balrampur Ch
ini, Wipro & HDFC Bank?', 'Hot Stocks | IPCA Laboratories, HDFC Life, Marico can give up to 33% retur
n in short term', 'Market LIVE Updates: Sensex, Nifty trade higher; mid, smallcaps outperform']
```

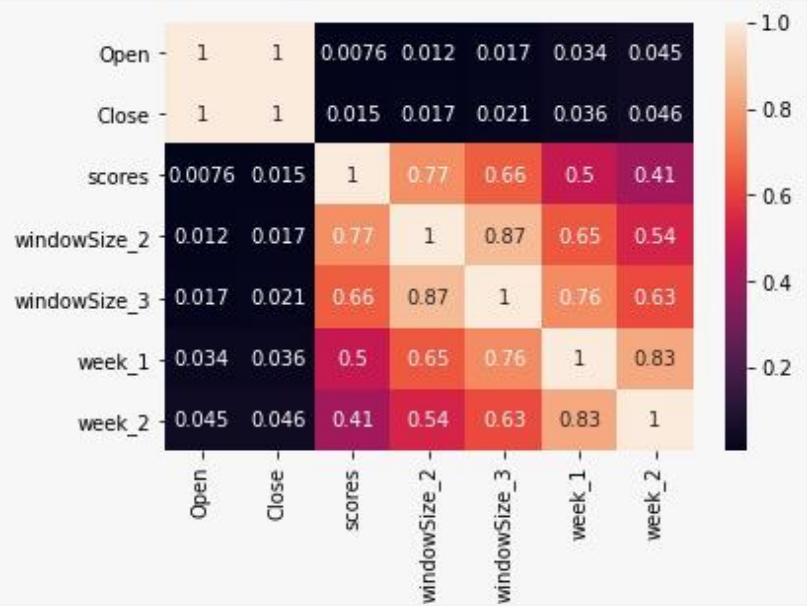
DATA PREPROCESSING

Analysing Results of VADER scores - Concluded its failure, looked for other models

1. Used VADER from nltk to loop over each headline contained in the respective news array. Compound scores were used.
2. For each date we took the average of the all the scores returned from all the headlines of that date.
3. Scores column w.r.t dates was added to dataset.
4. Correlation between the scores and the target>
5. ***Created variations in scores. Hope for better results***

In [6]:
```
scores=[]
for i in range(len(al)):
    score=0
    for j in range(len(al[i])):
        score+=sid.polarity_scores(al[i][j]).get('compound')
    avg_score = score/len(al[i])
    scores.append(avg_score)
print(len(scores))
```

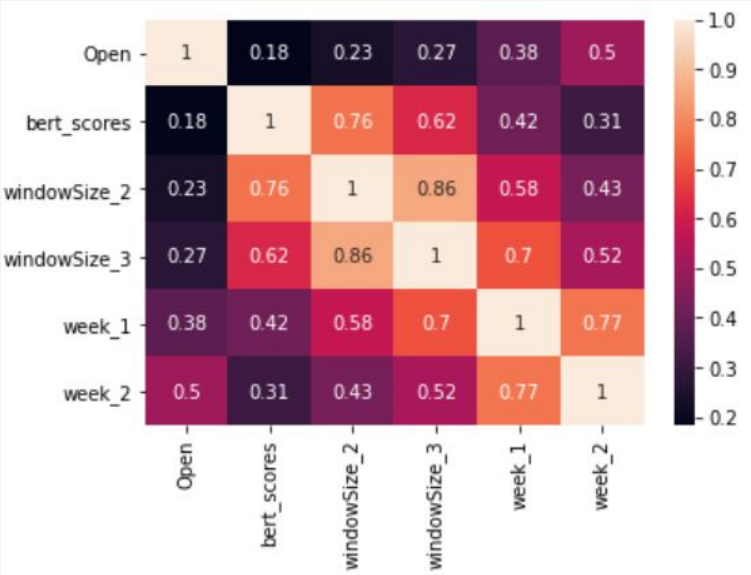
```
2552
```



DATA PREPROCESSING

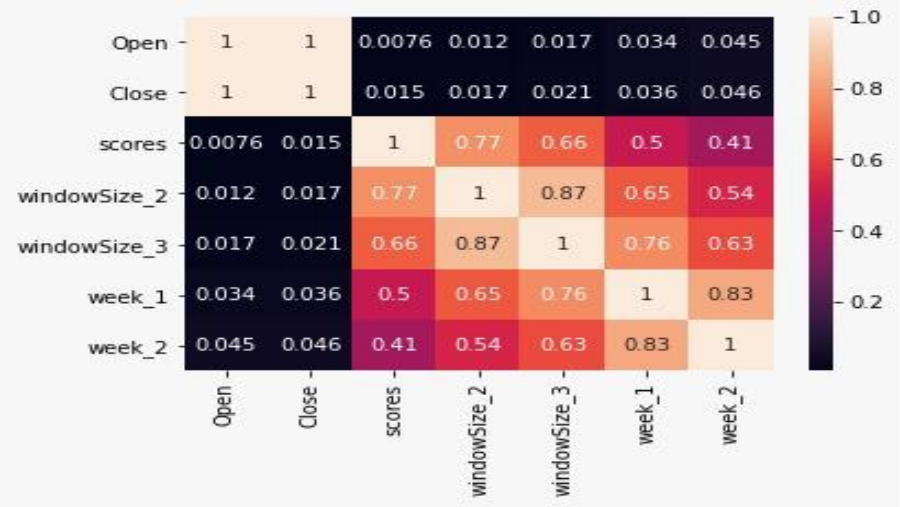
Analysing Results of FinBERT scores - Promising Results, explored weighted averages

1. Used FinBERT to loop over each headline contained in the respective array of the news array.
2. FinBERT : neutral : 0 ; Positive : +1 ; Negative - -1
3. Averaging similar to VADER(previous slide)
4. Scores column was then added to our dataset.
5. Correlation between the scores and the target.



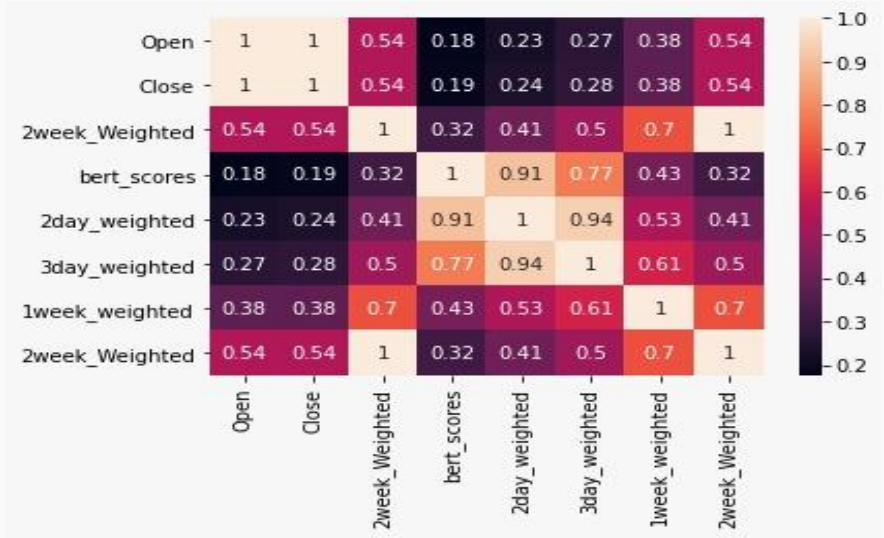
6. We experimented with the averaging technique by including weights. Here we gave higher weights to more recent news(asper dates) by describing a linearly decreasing function for deciding weights.
   a. Eg for 1 day lagged(total 2days new) : w[x] = 0.66 - 0.33x (x is the index)

ANALYSIS



VADER CORRELATION MATRIX

Low correlation - Column may not provide good information to mode while training



FINBERT CORRELATION MATRIX

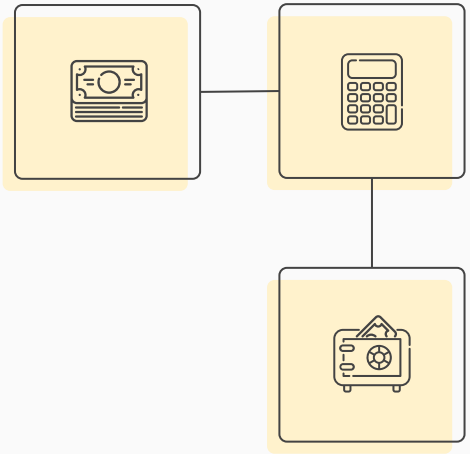
High correlation - Column might provide good information to model while training

Final Step - Dataset Ready

1. Weight averages gave even better correlation, hence added to dataset.
2. Observations and predictions with each NLP column have been separately documented and compared later.

TRAINING

SPLITTING THE DATASET



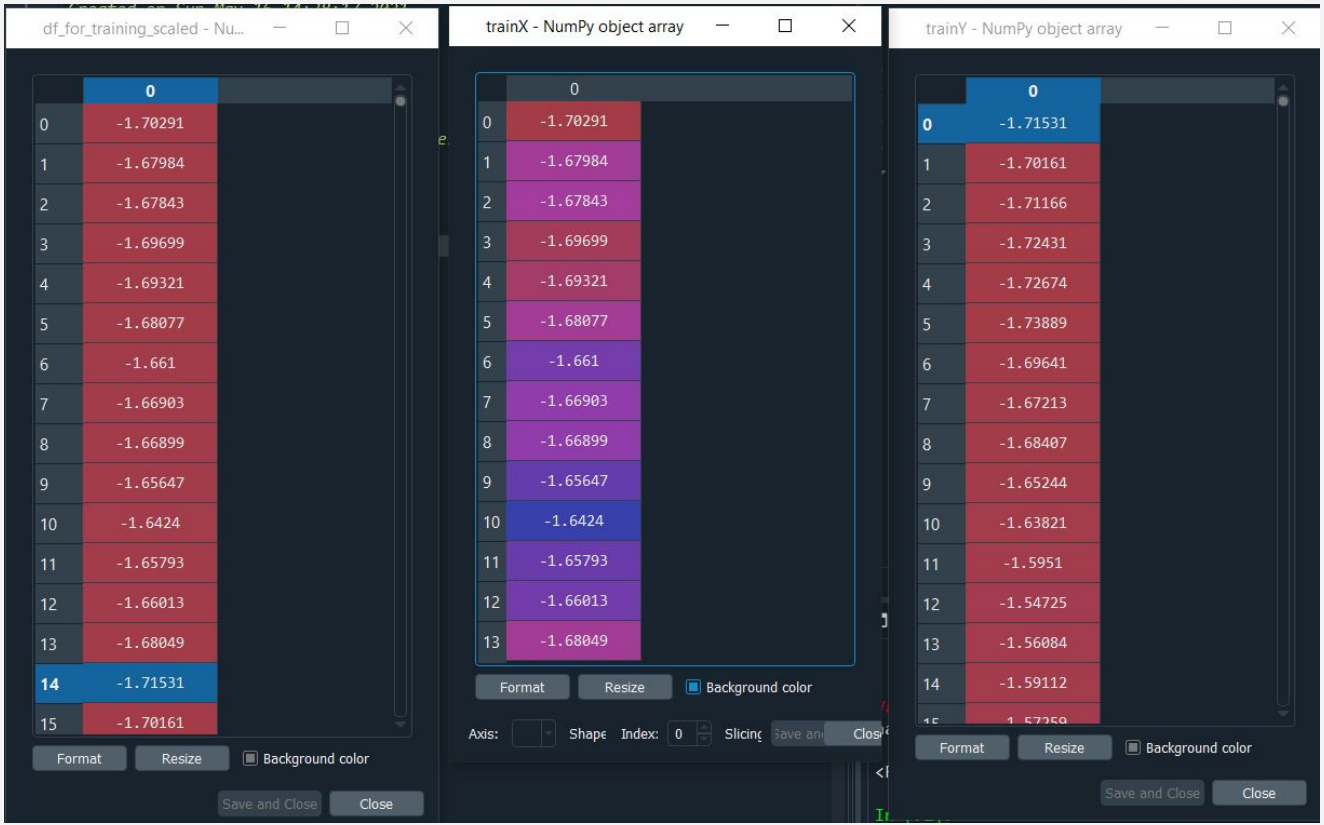
TRAINING THE MODEL
USING LSTM,GRU

PREDICTION

SPLITTING THE DATASET

- Data set is splitted into two categories one is test and train test from 2008 september to 2021 january. Test form 2021 february to 30 days after that.
- While creating the train and test data we are taking the window size of 14 and 7. It is like we are seeing the previous 14 days and predicting the 1 in this way the train x any train y for prediction is prepared . This same applies for 7 day window size.

LOOK BACK PERIOD



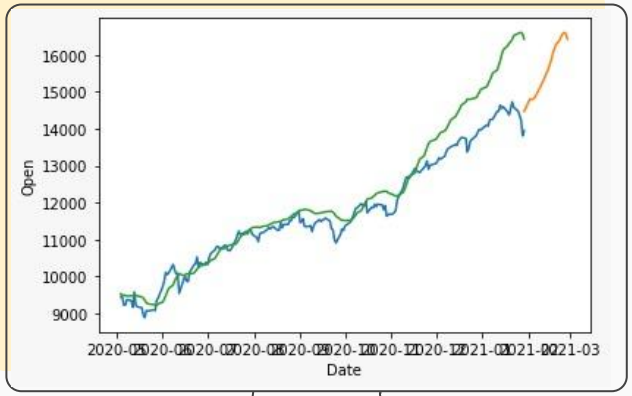
TRAINING THE MODEL USING LSTM,GRU

- Long short-term memory (LSTM) is a type of recurrent neural network that allows long-term dependencies in a sequence to persist in the network by using "forget" and "update" gates. It is one of the primary architectures for modeling any sequential data generation process, from stock prices to natural language.
- We are training our model on multivariate LSTM(Long short term memory)
- In our model we have used on layer of lstm with 120 units , and the activation function is relu . And a dropout layer is added of 0.3 this is used to reduce our model to over fit.
- After this layer we have a dense layer with one unit because we are only predicting the open price if the stock.
- The model is same for both LSTM and GRU.

CODE DEMONSTRATION

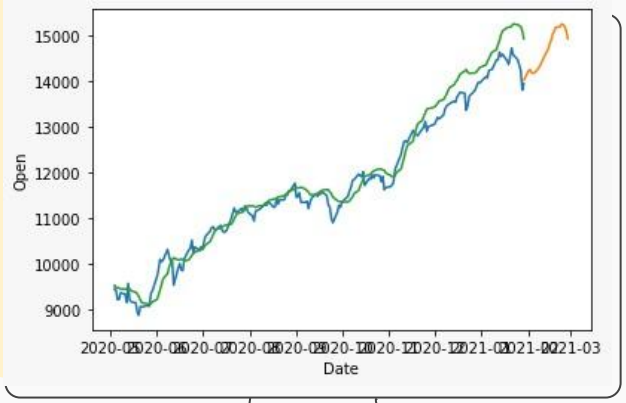
Blue - Actual,
Green - Model Training
Yellow - Forecasts

PREDICTIONS



WITHOUT SENTIMENT ANALYSIS

The LSTM model was trained and tested on the Dataset without taking sentiment scores into account.



WITH SENTIMENT ANALYSIS

The LSTM model was trained and tested on the Dataset with taking sentiment scores into account which was calculated by FinBERT

PREDICTIONS

1. LSTM(with and without sentiment)

- Predicted Window - 21 days
- Lookback size - 7 days

Type	MSE	MAE	MAPE
Without Sentiment	647977.2137757116	630.7099513809524	0.042435809556255706
With sentiment(2week)	756663.0971496409	678.6332712857144	0.04548751787897253
With Sentiment(1week)	467037.7502019805	569.1553612857145	0.03800686726217849
With Sentiment(3days)	254628.97642213182	430.3315279523806	0.028995393415229064

PREDICTIONS

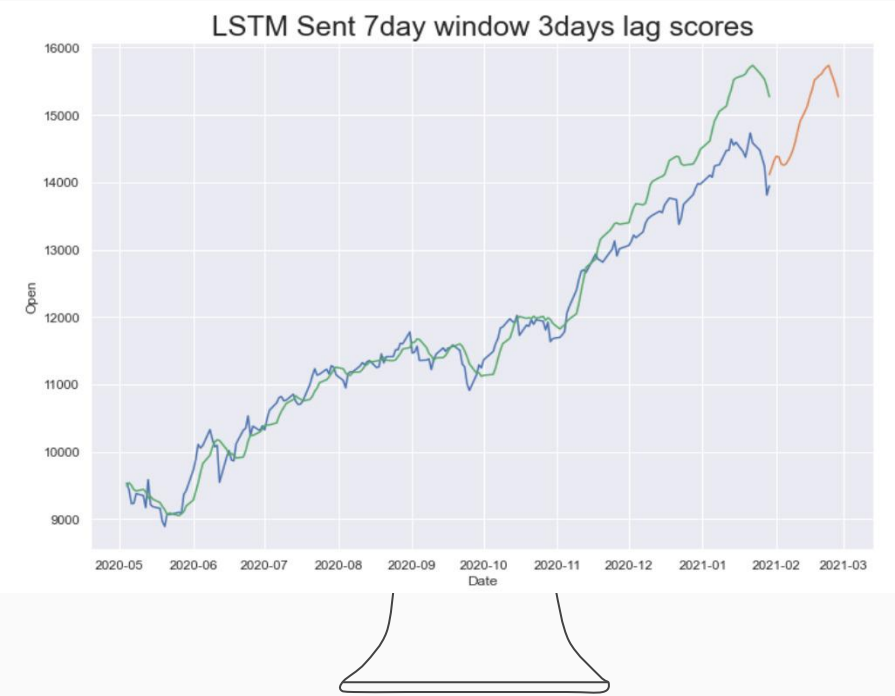
1. LSTM(with and without sentiment)

- Predicted Window - 21 days
- Lookback size - 14 days

Type of Data	MSE	MAE	MAPE
Without Sentiment	864577.3038522463	689.5893031904761	0.04643926711431763
With Sentiment(2week)	473333.5418729142	583.5811730000003	0.03889286277814990
With Sentiment(1week)	349845.01349752303	514.1932422380951	0.034349073970992086
With sentiment(3days)	373853.7378489153	548.2033560476193	0.03662869951104582

Blue - Actual,
Green - Model Training
Yellow - Forecasts

PREDICTIONS



PREDICTIONS

2. ARIMA

2. ARIMA

Model	MSE	MAE	MAPE
ARIMA	2238320.1017061346	1444.2212430000004	0.09622921569431611

PREDICTIONS

2. GRU(with and without sentiment)

- Predicted Window - 21 days
- Lookback Size - 7 days

Type	MSE	MAE	MAPE
Without Sentiment	1132471.8471608334	894.4148373809522	0.060190939471106084
With Sentiment(2week)	297595.21434753574	447.38433966666656	0.029905192841074863
With sentiment(1week)	823062.1190218959	814.5386491904762	0.054311121275039564
With sentiment (3day)	654733.847610565	697.7538872857147	0.046502510106130115

PREDICTIONS

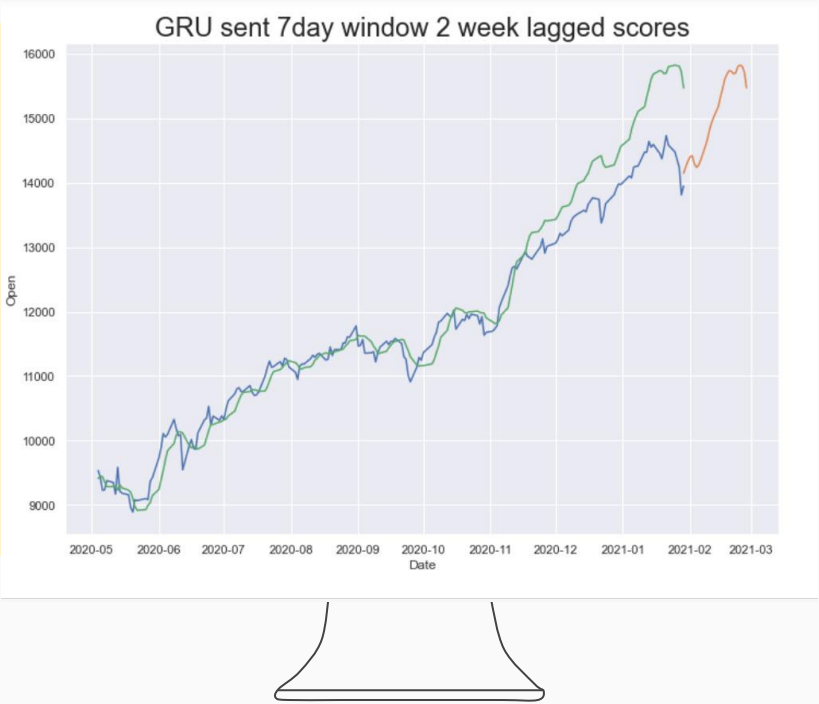
2. GRU(with and without sentiment)

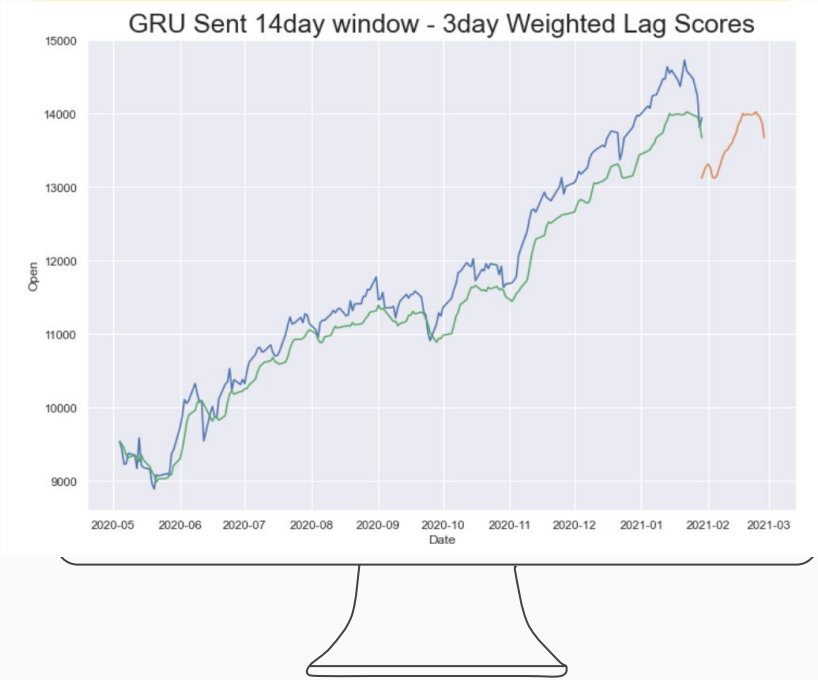
- Predicted Window - 21 days
- Lookback Size - 14 days

Type	MSE	MAE	MAPE
Without Sentiments	891751.0804768018	855.023553952381	0.05706105735121074
With Sentiments(2week)	700745.3886842157	741.2998292857145	0.04940032592062250
With sentiment(1 week)	738020.2481876051	756.2227206190479	0.05036917941496421
With Sentiment(3 day)	465670.8737423749	577.3670539523812	0.03847765215564279

Blue - Actual,
Green - Model Training
Yellow - Forecasts

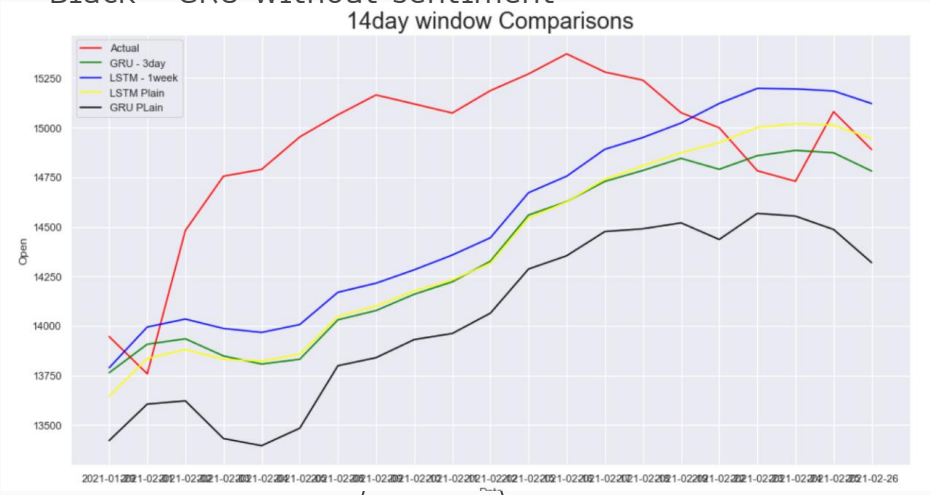
PREDICTIONS





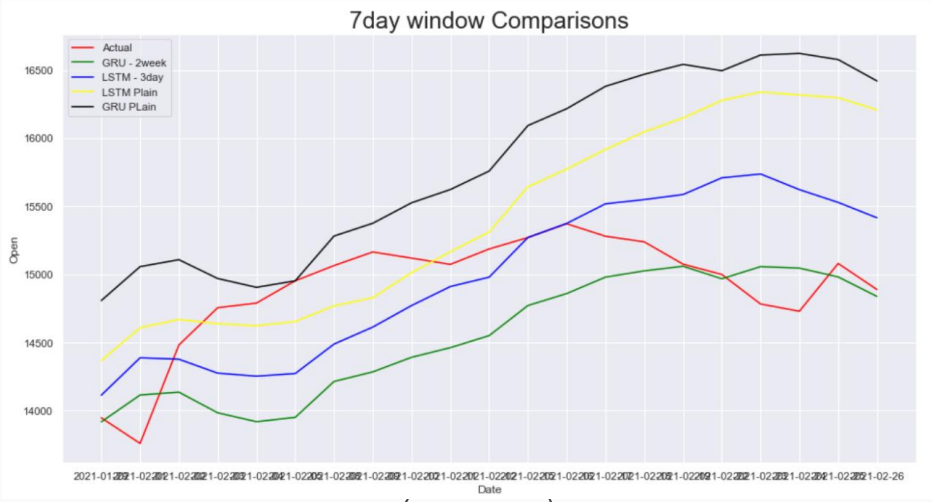
RED - Actual,
Green - GRU sentiment
Blue - LSTM sentiment
Yellow - LSTM without sentiment
Black - GRU without sentiment

COMPARISON OF MODELS



14 DAY WINDOW

LSTM 1 week gave better results compared to remaining models.



7 DAY WINDOW

LSTM 3 day and LSTM plain gave better results compared to remaining models.


THANK YOU