STOCK MARKET PRICE PREDICTION

Date - May 7, 2021
Mentor - Prof. Jaydip Sen

MEMBERS

ANANDA CHATTERJEE
HRISAV BHOWMICK
AVINASH KR YADAV
SANKALPA SAHA
VINEET KUMAR

Agenda

- Objective
- Data Description
- Methodology
- Time Series Models illustration
- ML Models illustration
- DL Models illustration
- Prediction using Sentiment Analysis
- Web App & Deployment
- Future Scope
- Conclusion

Objective

- Predict future value of the stocks of a company in such a way that for a given corpus of fund how one can effectively invest into different sectors by maximizing the profit while minimizing the risk.
- Understanding of Time Series, Machine Learning and Deep Learning methodologies for stock prediction.
- Use of above methods to predict on the current values of stocks by training on their previous values.

Data Description

- **Infosys data** was gathered from Yahoo Finance for the years of 2004 to 2019.
- The original data has 4026 rows and 6 columns such as open, high, low, close, adj close, volume.
- **Percentage of close** have been computed and added as a column name 'Returns' as it is required for few model buildings. So in total we have 7 columns and 4026 rows.
- Null values have been removed from 'Returns' column.
- Generally we have used 'Close' or 'Returns' column for prediction.
- Data from 2004-2018 was selected as train dataset while data of 2019 was selected for test dataset.

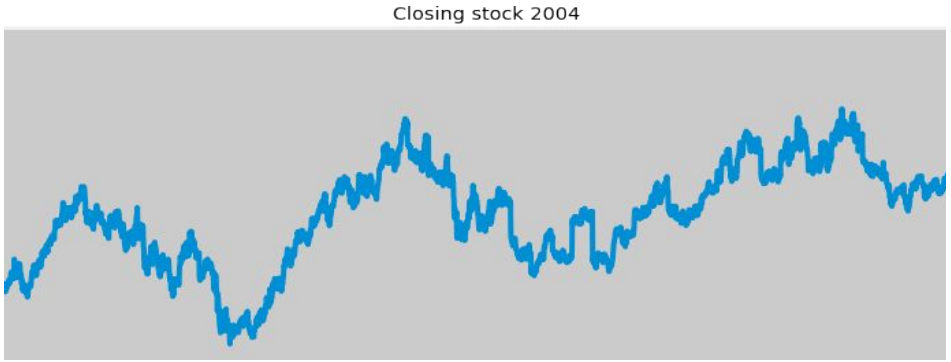
Methodology

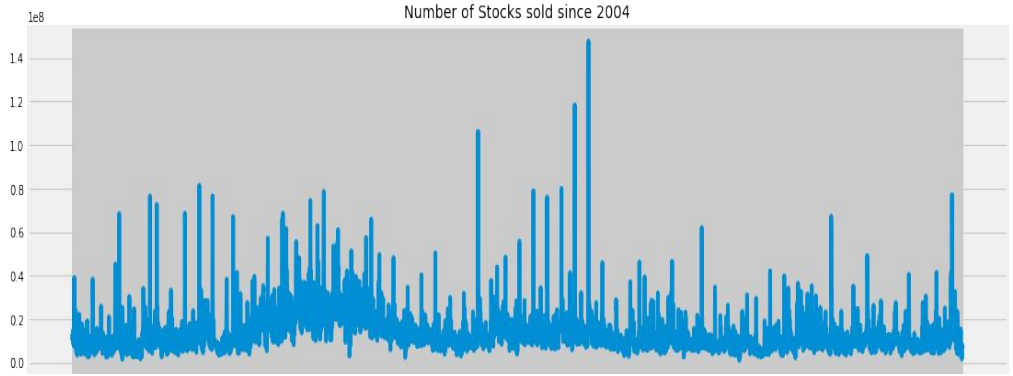
- A complete **Time Series** illustration has been done to forecast stock prices of 2019 while the model has been trained on data of 15 years (2004-2018). Total 7 Time Series model have been developed such as 'Simple Average', 'Moving Average', 'Exponential Smoothing', 'Holt Winter Exponential Smoothing', etc.
- 6 **Machine learning** models using Linear Regression, MARS, Random Forest, Gradient Boost, XGBoost, KNN have been built to predict the 'Close' value of stock price where low, open, high, volume, adj close were selected as the independent features. RMSE have been computed for all the models.
- **Classification** has also have been done on the 'Returns' column.
- **Deep Learning** models using Simple RNN, GRU, LSTM has been designed to predict stock prices where the model has been trained on previous 15 years data in daily basis and the model has been tested on data of 2019. Time step of 75 days was used for this.

Time Series Models

Time Series Illustration

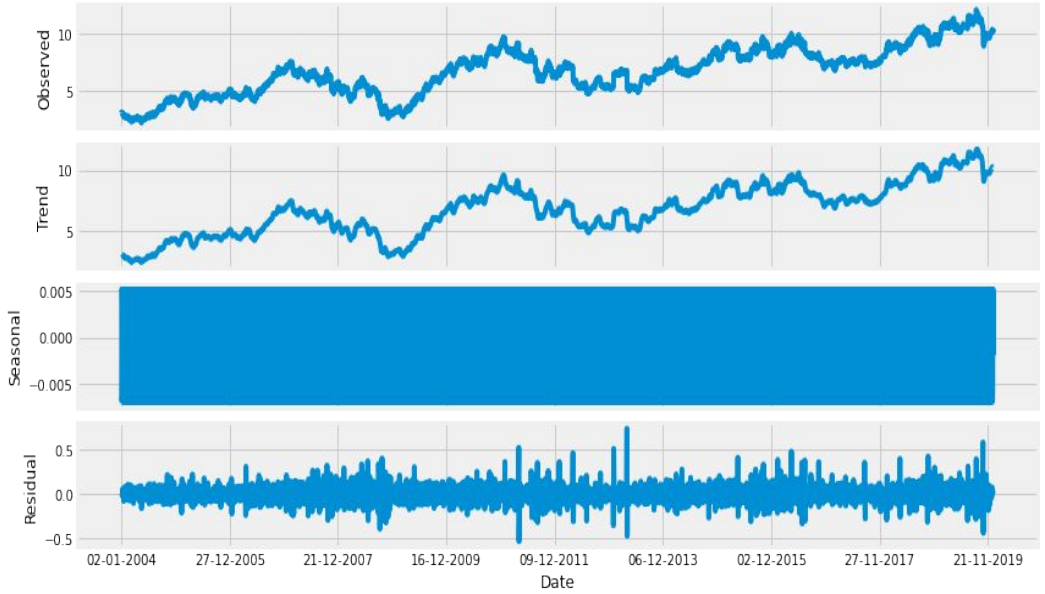
- Close value of stock since 2004 has been depicted.
- Volume of stocks have been depicted since 2004.
- Time Series was decomposed via additional seasonality.





Time Series Decomposition

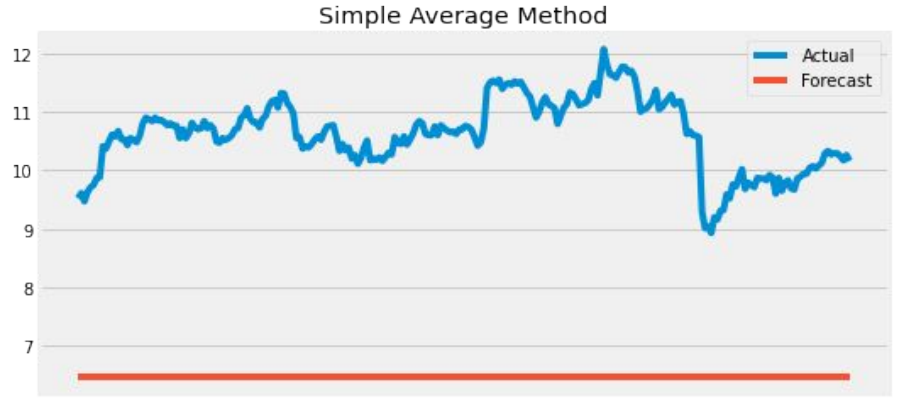
- Additive Seasonal decomposition has been done when we add individual components (Level + Trend + Seasonality + Cyclicity + Noise) to get the time series data.
- Constant seasonality has been found here on daily basis, and that's why Multiplicative decomposition has not been performed.



Simple Average & Moving Average Method

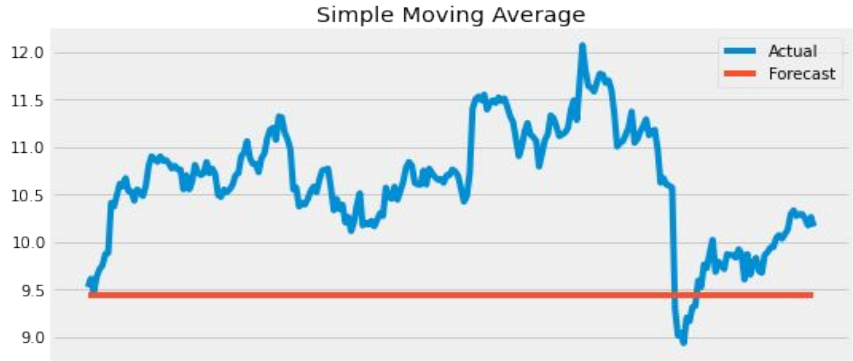
❖ Simple Average Method

- To predict the stock price of 2019 depending on the average values of 2018.
- RMSE Score came out as 4.1888



❖ Moving Average Method

- It does forecasting based on the average value of a moving window starting from the last observation in backward way. Here window size has been taken as 9.
- RMSE Score came out as 1.33



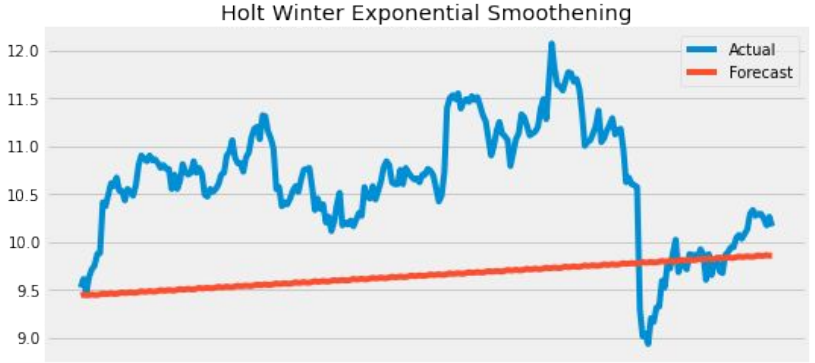
Exponential Smoothing & Holt-Winters Exponential Smoothing

❖ Simple Exponential Smoothing

- It forecasts without trend and seasonality.
- It forecasts using the weighted sum of the past observations where the weight decreases exponentially with past observations getting older.
- Here single parameter alpha=0.2 has been chosen as smoothing factor.
- RMSE score has come out as 1.3501 after forecasting using this model on the test dataset.

❖ Holt-Winter Exponential Smoothing

- It includes level, trend, and seasonality.
- RMSE score has come out as 1.1577 after forecasting using this model on the test dataset.

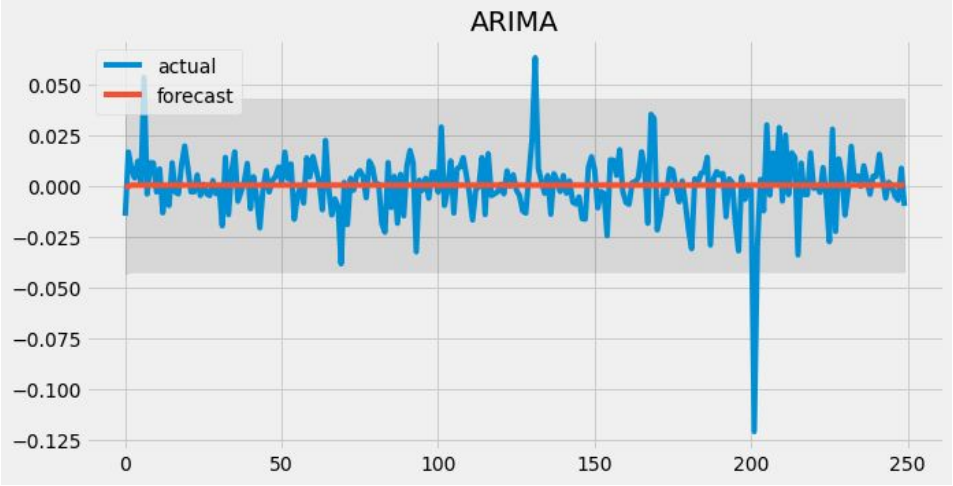


ADF & KPSS Test

- Both are used for **Stationarity checking** of the time series.
- Before applying the ARIMA model we need to know the difference value so we ran these two tests on the percentage of the close column here it is named as ‘Returns’.
- The p-value for **ADF test** obtained as much less than 0.05 which rejects the NULL Hypothesis, i.e the series is stationary.
- The **KPSS test** verified this result with a p-value of 0.1 which accepts the null hypothesis, i.e the series is stationary.

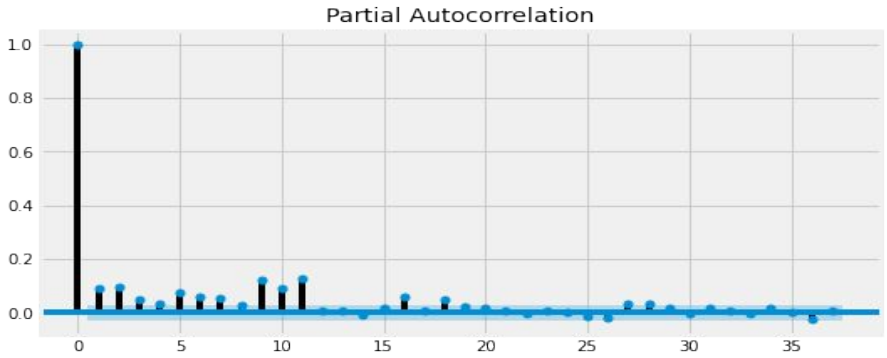
ARIMA Model

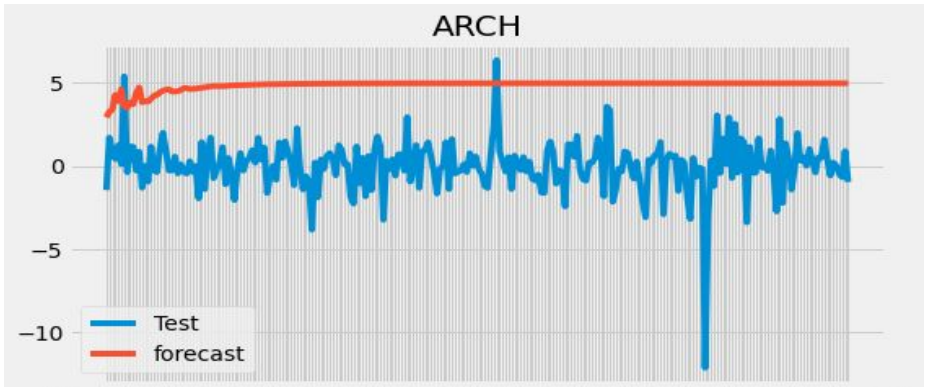
- ARIMA model consists of 3 parameters as **p, d and q** where p is the highest lag in Regression model, d is the order of differencing to make the series stationary and q is the num of past error terms included.
- In our case as no differencing was required, so d was 0.
- To find out the ultimate value of p and q **auto_arima** function was applied.
- The ultimate value of p, d and q yielded as 0, 0, 2 with the AIC value -19515.113.
- RMSE Score of ARIMA came out as 0.0153.



ARCH Model

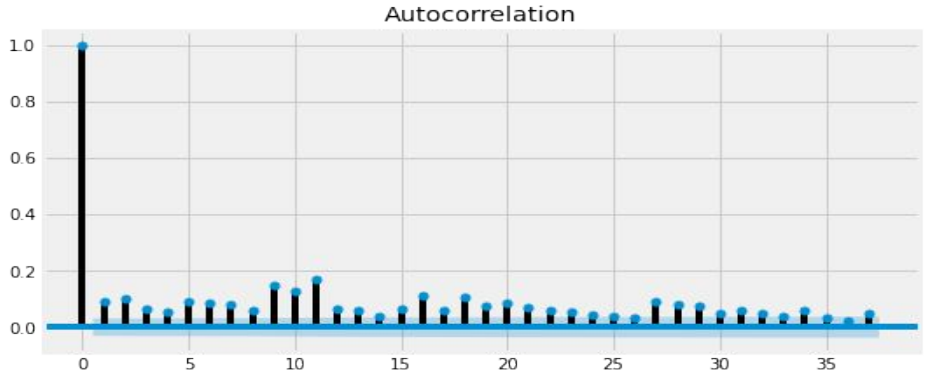
- It gives a measure of volatility which indicates how likely the stock holds good.
- ARCH is used to know the **conditional variance** of the Residuals over a time period.
- The p-value of ARCH model is 18 which can be observed from pacf graph but with this p-value only alpha(1) and omega was coming out as significant. P-value of all others alpha was >0.05. So we took p=1 which results in alpha(1) and omega significant.

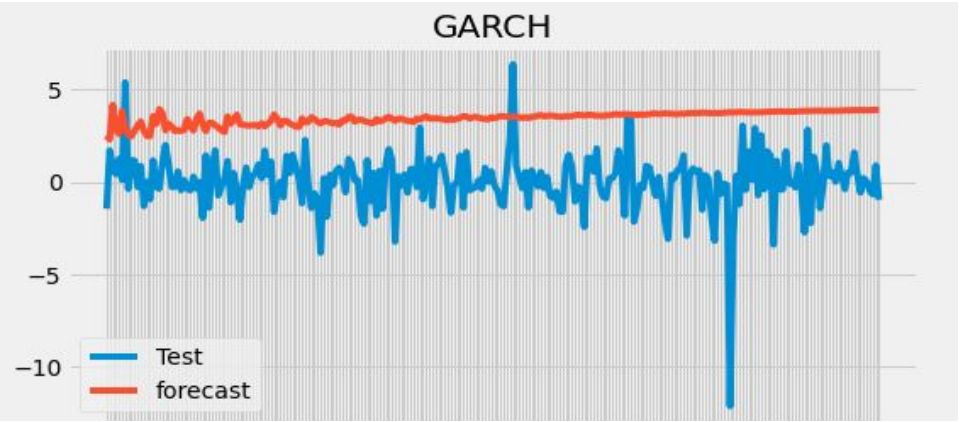




GARCH Model

- GARCH includes the **past squared observations along the past square variances** to model the variance at time t.
- Both ARCH and GARCH was applied on the percentage of close columns which is named as 'Returns'.
- The q value of the GARCH is 37, which can be seen from the acf plot but with this p value all the coefficients except three were coming out as insignificant so we took q=1 with this all the coefficients alpha(1), beta(1), and omega came out as significant.



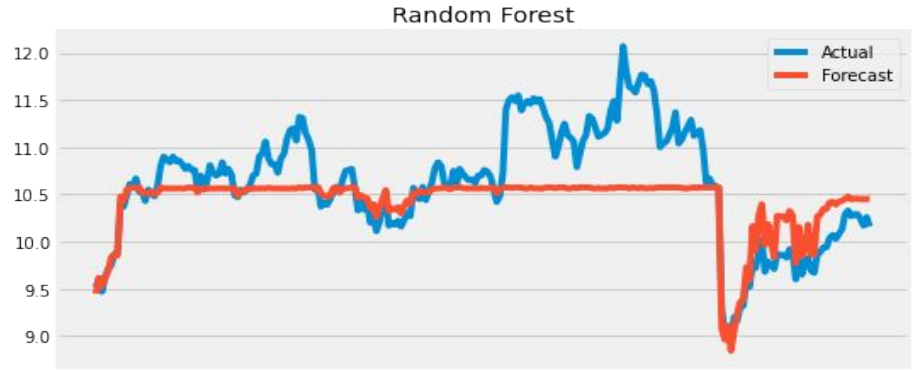


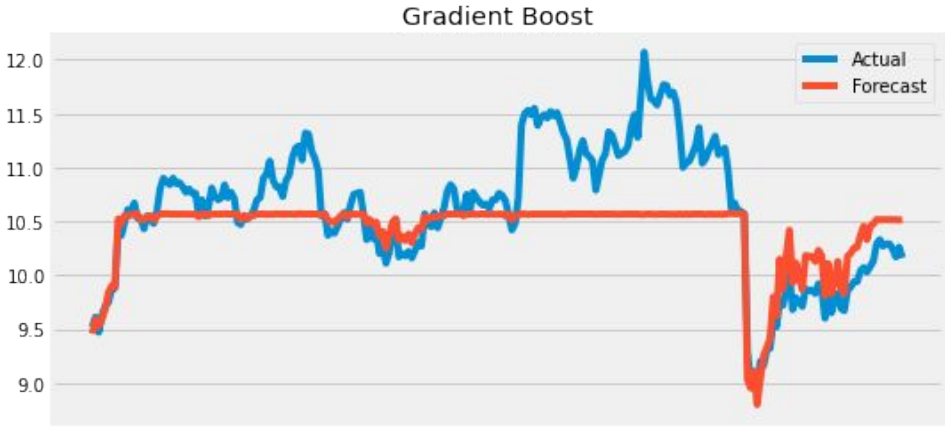
Machine Learning Models

Random Forest & Gradient Boost Regressor

- Data from 2004-2018 was taken as training data set and data of 2019 was taken as testing data.
- The target column is Close and others are independent variables.

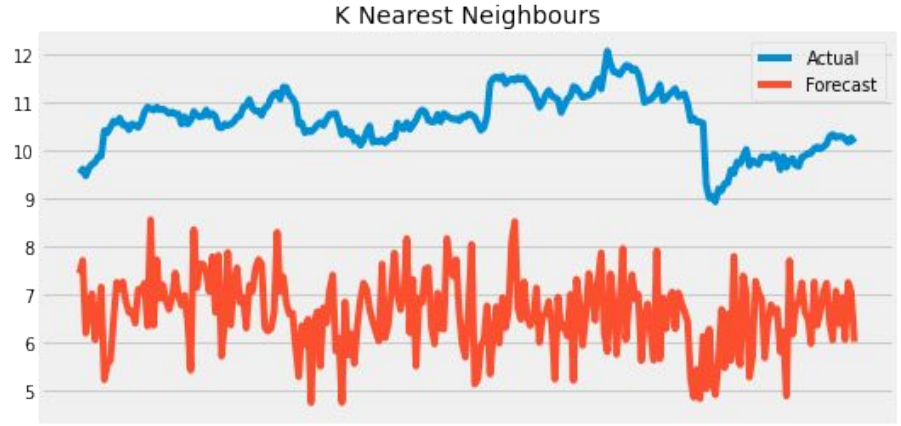
ML Models	RMSE Score
Random Forest Regressor	0.4518
Gradient Boost Regressor	0.4509





XG Boost & KNN Regressor

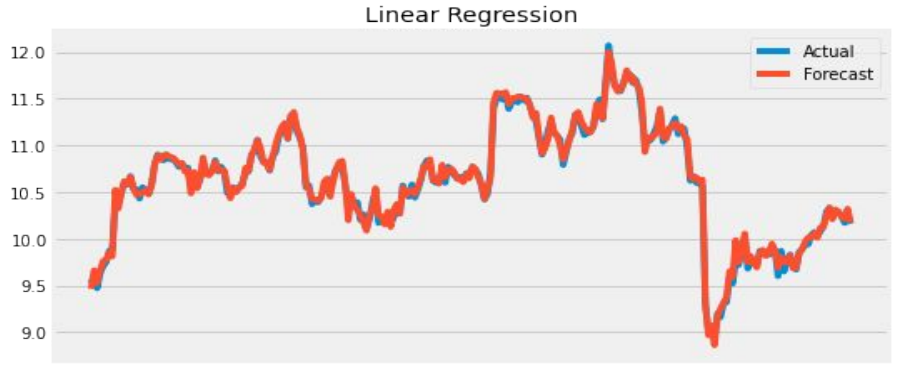
ML Models	RMSE Score
XGBoost Regressor	4.087
KNN Regressor	4.098

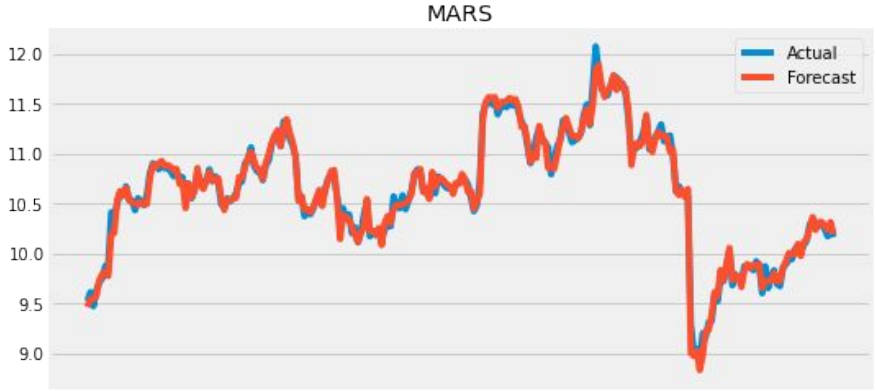


Linear Regression & MARS

- MARS Regression model is used when there are more than one inputs and the target variable shows non-linear relationship with the input variables.

ML Models	RMSE Score
Linear Regressor	0.046
MARS	0.064





Machine Learning Classifiers

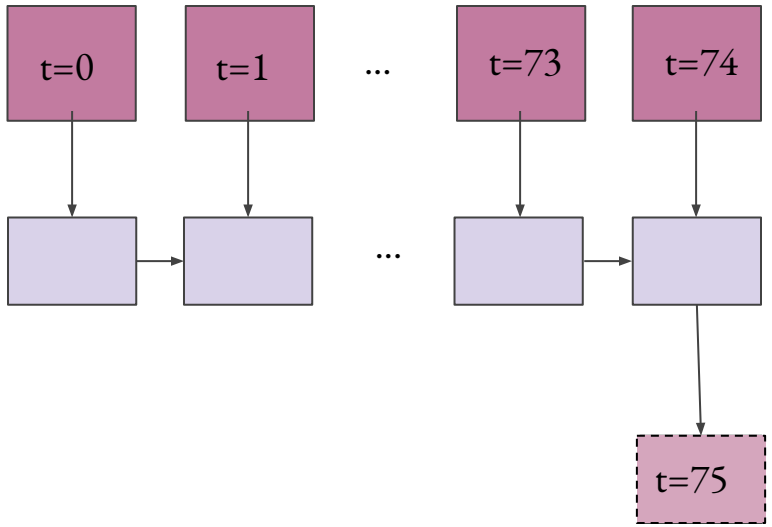
- The output of the **Percentage of Close** column gives output in both positive and negative values, those positive values was encoded as 1 and negative values was encoded as 0 and those 1 and 0 were stored in a separate column named 'CL'
- In Classification also the data from 2004-2018 was considered as train data and data of 2019 was considered as test data.
- The **target column here is 'CL'**.
- **Pycaret** library was run which comprises of all the classifiers so we can compare output of different classifiers.

	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC	TT (Sec)
dt	Decision Tree Classifier	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	0.019
rf	Random Forest Classifier	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	0.545
ada	Ada Boost Classifier	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	0.019
gbc	Gradient Boosting Classifier	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	0.207
lightgbm	Light Gradient Boosting Machine	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	0.090
et	Extra Trees Classifier	0.9972	0.9999	0.9960	0.9987	0.9973	0.9943	0.9943	0.509
ridge	Ridge Classifier	0.9606	0.0000	0.9885	0.9403	0.9637	0.9207	0.9224	0.019
lda	Linear Discriminant Analysis	0.9606	0.9966	0.9885	0.9403	0.9637	0.9207	0.9224	0.019
qda	Quadratic Discriminant Analysis	0.9294	0.9917	0.9872	0.8909	0.9365	0.8574	0.8634	0.019
nb	Naive Bayes	0.5289	0.5118	0.8915	0.5315	0.6659	0.0175	0.0269	0.017
lr	Logistic Regression	0.5268	0.4796	1.0000	0.5268	0.6901	0.0000	0.0000	0.298
knn	K Neighbors Classifier	0.5026	0.5083	0.5485	0.5266	0.5370	0.0001	0.0001	0.122
svm	SVM - Linear Kernel	0.4956	0.0000	0.4000	0.2112	0.2764	0.0000	0.0000	0.023

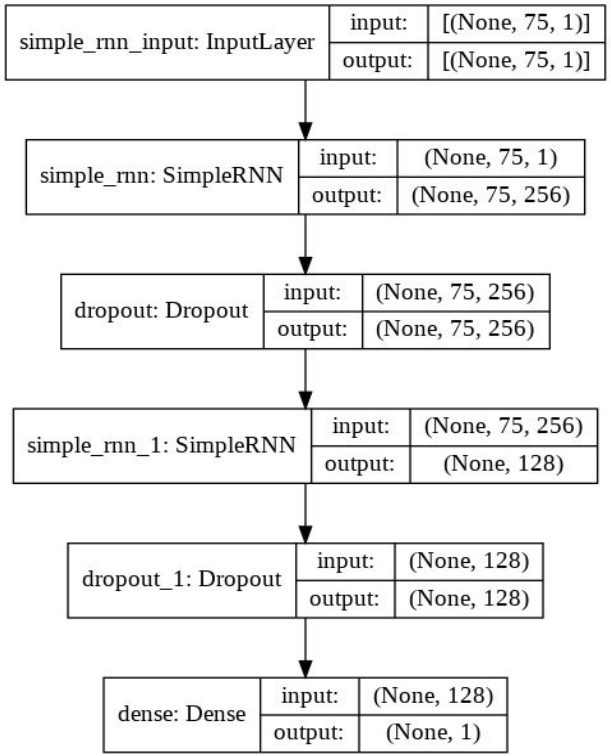
Deep Learning Models

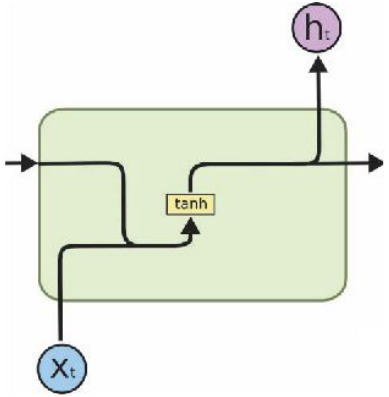
Time Steps

- Data is divided into train set (2004 to 2018) and test set (2019).
- 75 days **rolling window** is used as time step.
- Each of the Close value is predicted based on previous 75 days trained data.
- A model will be used to make a forecast for the time step, then the actual expected value from the test set will be taken and made available to the model for the forecast on the next time step.
- The model was compiled with the help of **Adam optimizer** and the error is computed using **RMSE**.
- The network is trained for 100 epochs with a batch size of 64.



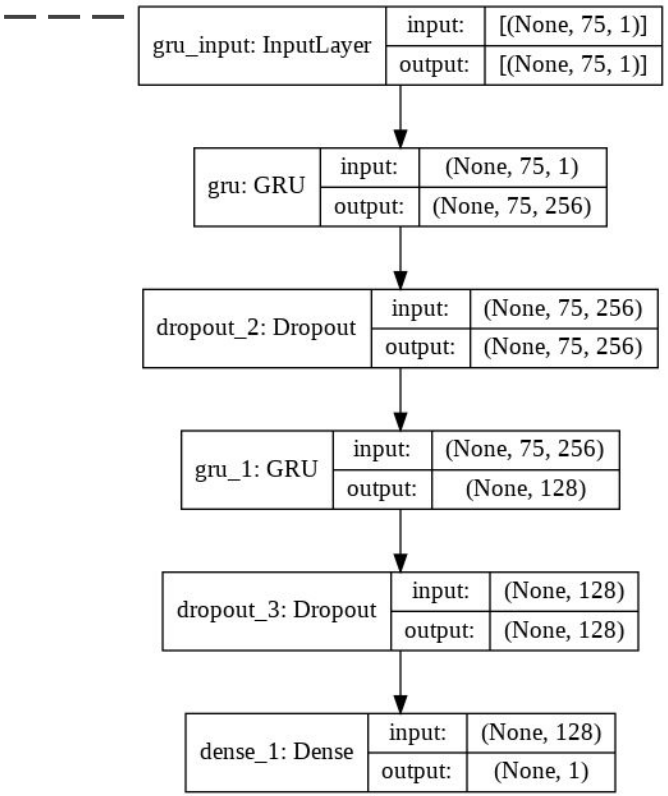
Simple RNN

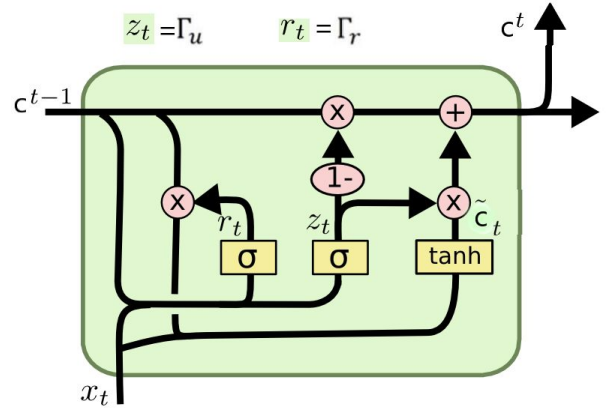




- The input layer of LSTM has num of time steps = 75 and num of features = 1.
- 1st hidden RNN layer has 256 nodes, followed by a 20% Dropout.
- 2nd hidden RNN layer has 128 nodes, again followed by a 20% Dropout.
- And finally a dense output layer.

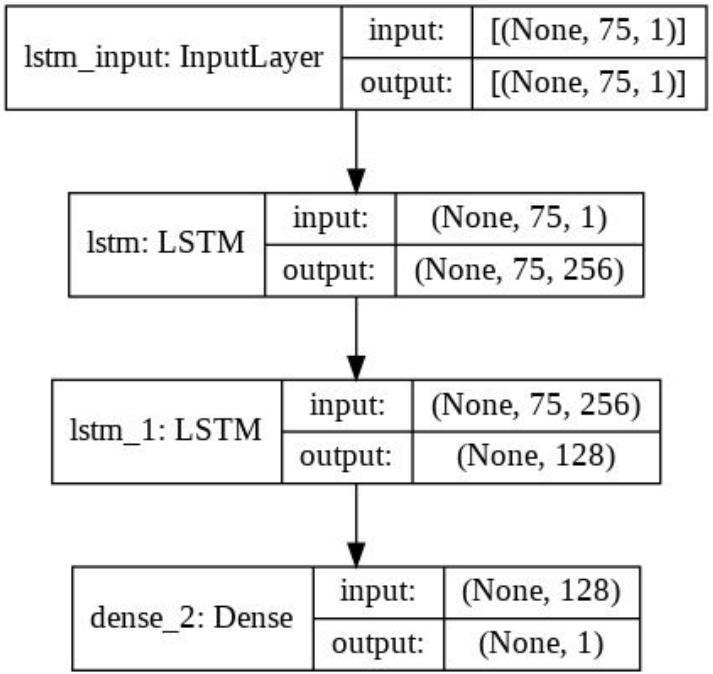
GRU

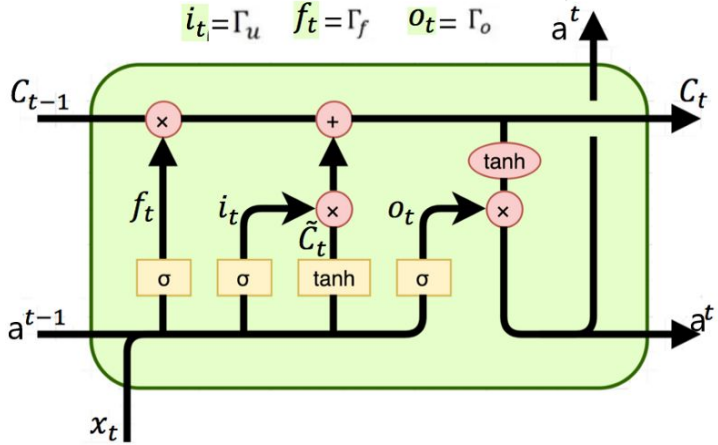




- The input layer of GRU has num of time steps = 75 and num of features = 1.
- 1st hidden GRU layer has 256 nodes, followed by a 20% Dropout.
- 2nd hidden GRU layer has 128 nodes, again followed by a 20% Dropout.
- And finally a dense output layer.

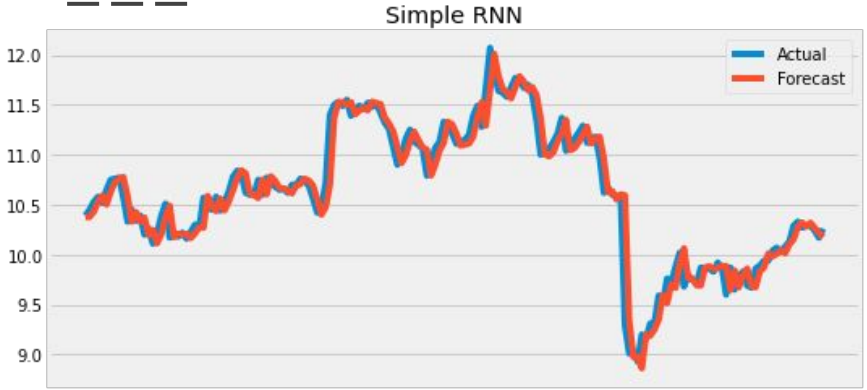
LSTM

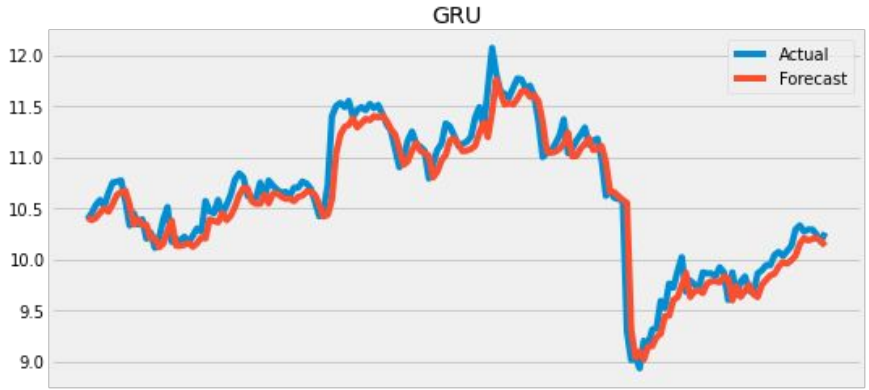


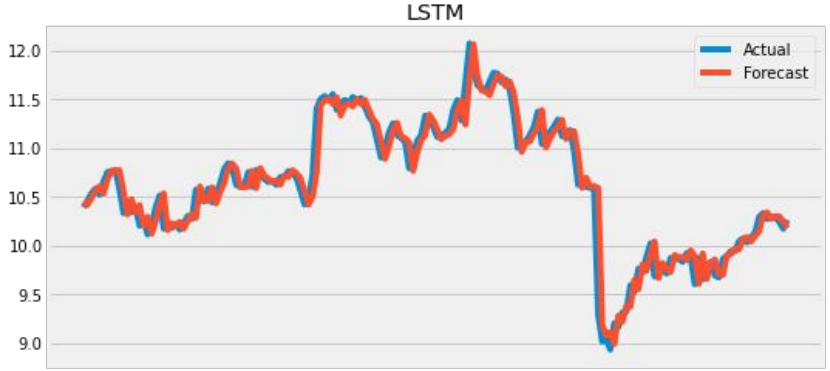


- The input layer of LSTM has num of time steps = 75 and num of features = 1.
- 1st hidden LSTM layer has 256 nodes.
- 2nd hidden LSTM layer has 128 nodes.
- And finally a dense output layer.

Test set Prediction



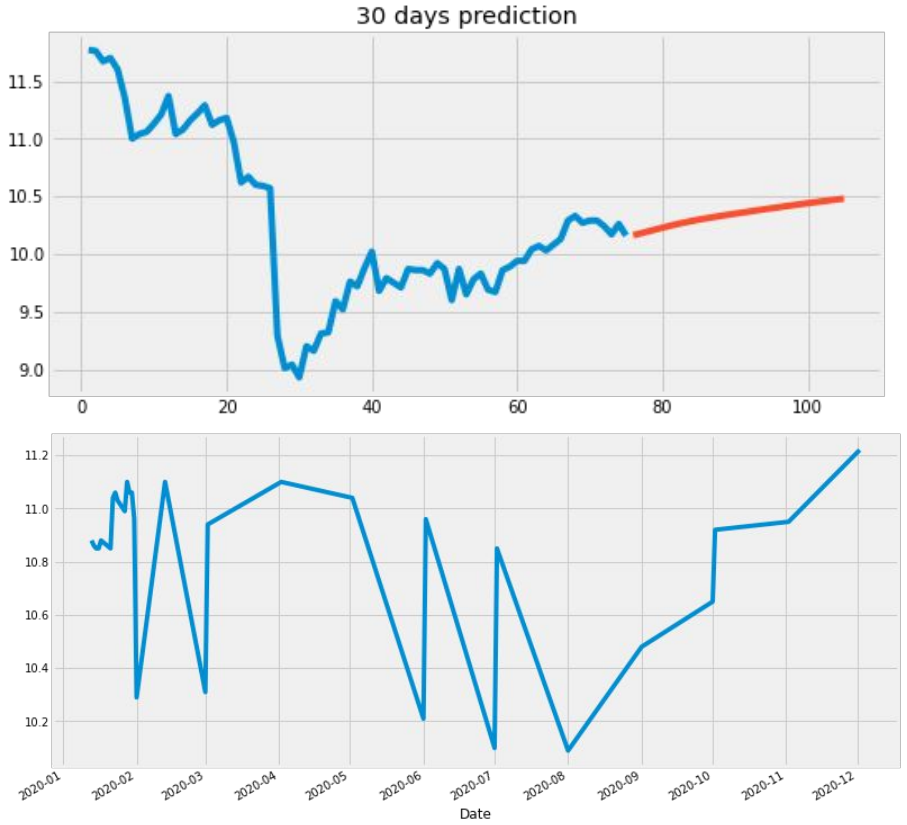




DL Models	RMSE Score
Simple RNN	0.1763
GRU	0.1982
LSTM	0.1713

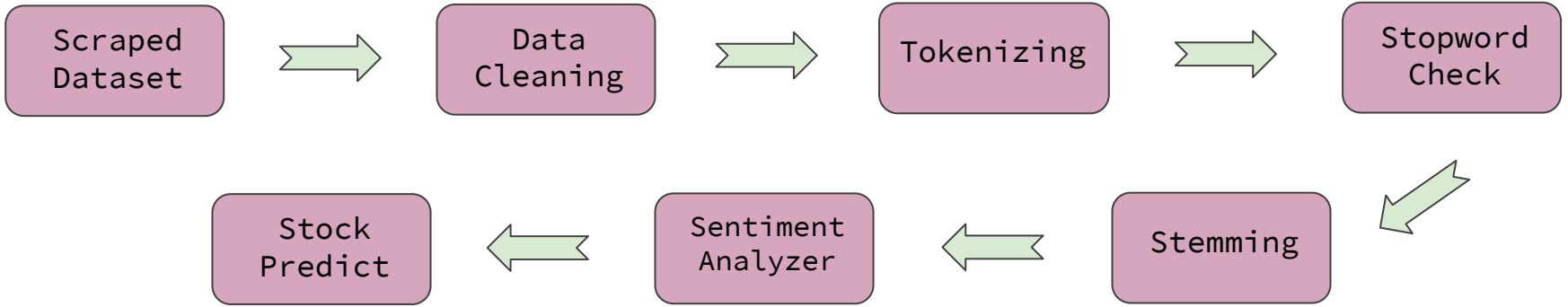
Future Stock Prediction using LSTM

- After the model is ready on train and test set, now we are predicting for **upcoming 30 days** which is not present in dataset.
- The prediction is shown in red on the top graph.
- We have tallied this data with **original 30 days** data of INFY which is shown in the below graph. As predicted, it depicts a upward trend of stock price.



Sentiment Analysis
for Stock Price Prediction

Process Workflow

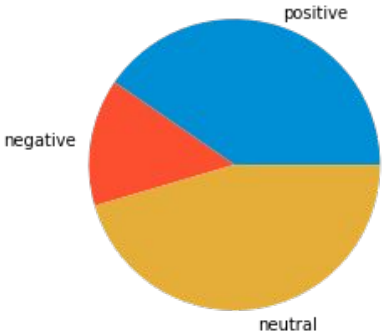


- Here's the dataset, which has Infosys news headlines from Jan 2009 to Jan 2018 ...

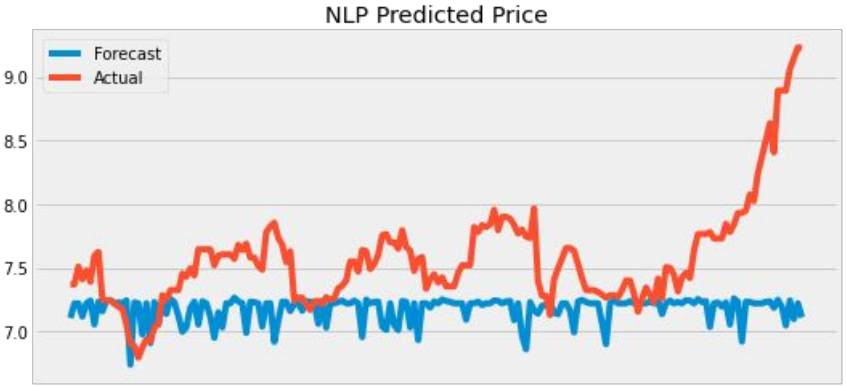
time	date	Source	News Headlines
3.38 pm	2009-01-01	Moneycontrol.com	Infy Q3 PAT seen at Rs 1572 cr: KRChoksey
9.05 am	2009-01-01	Business Line	Corporate houses seek CISF protection
8.44 pm	2009-01-05	Moneycontrol.com	Hold Infosys Tech, TCS, Satyam: Emkay Global
1.00 pm	2009-01-05	Moneycontrol.com	Infosys' Dec qtr PAT seen at Rs 1491 cr: Angel
8.53 am	2009-01-06	Business Line	IT majors may miss Q3 revenue forecast: CLSA

Prediction using Sentiment

date	cleaned	Close	Compound	Negative	Neutral	Positive
01-01-2009	infy q3 pat seen at rs 1572 cr krchoksey corpo...	3.24875	0	0	1	0
05-01-2009	hold infosys tech tcs satyam emkay global info...	3.24875	0	0	1	0
06-01-2009	it majors may miss q3 revenue forecast clsa in...	3.32500	-0.1531	0.052	0.948	0
07-01-2009	satyam scam infosys peers react security fears...	3.36750	-0.6249	0.363	0.503	0.134
08-01-2009	infy dec qtr pat seen at rs 5607 cr religare w...	3.33000	0.3612	0	0.884	0.116

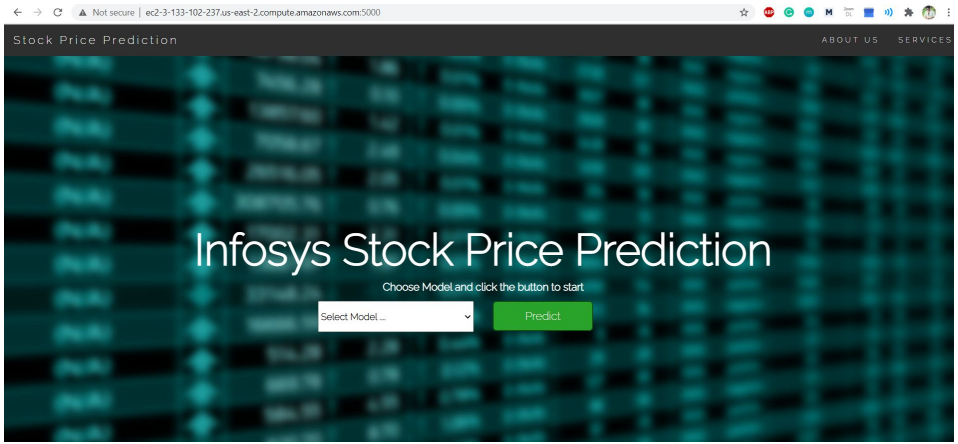


- Using **Sentiment Intensity Analyser**, we could find out the positive, negative and neutral sentiment.
- Based on the sentiment scores and the Close price, a model was trained on 2009-16 data using **Linear Regression**, and prediction was done on 2017-18 data.



Model Deployment

Web App



- UI has been built to visualize how the models are performing on 2019 test data.
- Finally, it has been deployed using Amazon-EC2.
- It can be accessed here - http://ec2-18-221-244-29.us-east-2.compute.amazonaws.com:5000/

Summary

Models	RMSE
Simple Average	4.189
Moving Average	1.333
Simple Exponential Smoothing	1.350
Holt Winter Exponential Smoothing	1.158
ARIMA	0.015
ARCH	5.061
GARCH	3.781
Random Forest	0.452
Gradient Boosting	0.451
XG Boost	4.087
K Nearest Neighbors	4.099
Linear Regression	0.046
MARS	0.065
Simple RNN	0.176
GRU	0.198
LSTM	0.171

- Among Time Series models, **ARIMA** gave us the best RMSE score.
- Among Machine Learning models, **Linear Regression** gave us the best RMSE score.
- Among Deep Learning models, **LSTM** gave us the best RMSE score.
- As LSTMs are widely used for sequence prediction problems and have proven to be extremely effective, we went ahead with it to predict stock values for INFY of 2020.

Conclusion

- We have tried to understand how different models perform on stock prediction. Moreover how the pattern of stock varies depending on the past values have also been observed from the output of different models.
- Using all the Time Series, Machine Learning and Deep Learning models we predicted on seen data. Only using LSTM model, we predicted stock values of 2020.
- Hence, we will be able to guide at which time one should invest in stock to gain maximum return.

Future Scope

- Tweaking some models, we can train the model to predict on different stocks of different sectors.
- SARIMA and SARIMAX can be applied on the data for forecasting if there is a presence of seasonality in the data.
- Instead of using only LSTM, we can use CNN to extract features from the data and then, we adopt LSTM to predict the stock price .
- We can use Reinforcement Learning also for Stock Prediction.
- Make investment portfolio with various sectors and customise it for different individuals.

Thank You