BIRLA INSTITUTE OF TECHNOLOGY & SCIENCE, PILANI
Work Integrated Learning Programmes Division
SEMESTER II, 2021-22
M.Tech.in Data Science and Engineering

Stock Recommendation System

By
(Arun kumar Ashok Gupta)
(2019AP04010)

On
28th Jan, 2022

Under the supervision of
Prof. V S VASAN, BITS

&

Under the mentorship of
Sibanand Sahoo, Lead data scientist

BIRLA INSTITUTE OF TECHNOLOGY & SCIENCE, PILANI
Work Integrated Learning Programmes Division
SEMESTER II, 2021-22
M.Tech.in Data Science and Engineering

DSE CL ZG628T DISSERTATION

Stock Recommendation System

Acknowledgement

I would like to express my special thanks to my project in-charge, **Prof. V S Vasan** as well as my mentor, **Mr. Sibananda Sahoo**, who gave me the opportunity to do this insightful project (**Stock Recommendation System**), which also helped me in doing a lot of Research and it provided me an opportunity to reflect on various new things.

I am really thankful to them.

Secondly, I would also like to thank my parents, my wife and friends who helped me a lot in finishing this project within the limited time.

I am making this project not only for marks but to also upgrade my knowledge.

THANKS AGAIN TO ALL WHO HELPED ME.

STAY SAFE.

Abstract

This project demonstrates the stock analysis tool as a ***Stock Recommendation System*** using data mining techniques, along with some deep learning tools for model preparation and data visualization. The recommendation is implemented by analyzing the association among the different stocks.If a stock symbol is provided as an input, the system will return a group of stocks which are similar in trend to original input stock. The comparison is done on the basis of stock association, same feature based grouping and similar price variation trend. Recommendations are also prepared by regression modeling. The system also contains a forecasting tool with built in seasonal modeling to reflect the future selected stock price range.

Our system also helps the users to display the top 5 highest performing stocks based on their consistent performance. Also like stocks, which have the same trend as the selected stock have. These results will provide useful insight to develop a more advanced version of the recommendation system that a traders.

Also this system provides the data, on which the technical analysis becomes very easy for the user. Moving averages over 100 days and 200 days are displayed along with historic stock prices on the tool.

Table of Contents

1. Objective

The objective of this project is to create a recommendation system for the Stocks based on historical data. As a stock recommendation system, this application should provide easy and understandable solutions to users that help them to make clear decisions on trading the stocks. Especially those who are new to stock trading. Also, it will help them to analyze the various aspects of the selected stock and do the technical analysis.

2. Introduction

a. Statement

The idea of this project is to generate constant or consistent profit from the stock transactions, which is a very challenging task due to the unpredictability of the market & changing day-to-day conditions which impact market sentiments and details of the companies performance.

This application is solving the problem, by helping the investors, who may be new to stock investment, to identify healthy stocks / groups of stocks. Our application will get the trends of the selected stock. Our application enables the investors to group up the stocks based on the various important aspects of the stocks. i.e *Price, Market cap, volume, PE Ratio, EPS, and Beta*. Also, it provides the group of similar stocks with selected stocks on the different accuracy level. This accuracy level is nothing but the size of the cluster on the selected above features.

This application also helps investors by providing the information of the top five most performing stocks and provides similar stocks, based on the selected stock.

Being the stock data is time series data by nature, we have also analyzed the data and created a time series model to predict the future selected stock price range, with accuracy and error indication.

Identifying healthy businesses or groups of businesses in a way is similar to trying to predict how a business will do in the future. The main goal of our system is to identify healthy publicly traded businesses (or associated businesses); a secondary goal is to give users enough information needed to invest and have a method to mitigate risk.

b. Benefits from this project

There is so much information that keeps on generating everyday. Similarly, everyday tons of information are kept on generating on Stocks. The daily news, economic reports, technology journals, it's all constantly being uploaded to the web. We all have ways of filtering our daily news feeds from visiting our favorite news sites to following different people on Instagram.

Similarly, users, who are new to stock trading, must have a method or tool, which will filter and identify the similarity based on the selected stocks. The stock market is a very tempting place to make money, but making wise decisions really matters. Our application will help them to understand the stock market and provide them with a piece of helping information to make trading decisions. Developing this application to sort, filter, and extract useful insight from historical raw data. Insight into the state of our economy helps everyone make better choices about their purchases and may even influence daily spending habits, as well as identify investment opportunities.

c. Background knowledge

Recommending the stock market performance usually depends on 3 analysis

1. Fundamental Analysis

Fundamental Analysis is a very involved process; the analyst must know a lot about the particular business they are interested in investing in. A fundamental analyst would analyze a business's profit margins, earnings, losses, direct competitors, production capabilities, even the management and employees.

2. Technical Analysis

Technical Analysis differs by predicting stock performance by mostly analyzing past market performance data; this method is also involved but it does not rely on understanding the fundamentals of a business.

3. Machine Learning

The latest methods for predicting stock prices are in the form of Machine Learning; from simple algorithms to Artificial Neural Networks. Being the stock market data are time series in nature, Recurrent neural network is suitable for this.

The stock market is very complex and convoluted for a beginner. Political events, market news, quarterly earnings reports, international influence and conflicting trading behavior, all have a direct impact on how it performs. A lot of studies have shown that predicting stock market returns is a difficult task. A stock is a type of security, which represents ownership in a company, its assets, earnings and dividends. It is an indicator of how the company is performing and what its growth/profits are. Stocks are traded in the stock market, where the prices are controlled by traders' bids (buy price) and offers (sell price). Stocks should be recommended to the investor based on his interests, preferences and trading behavior.

We plan to make this task less consuming and easily understandable to the user and provide solutions to the same.

3. Project Plan

#	Task	Names of Deliverables	Status	Comment
1	Data Acquisition	Data collection from various API and sources.	Done	Stock data has been collected.
2	Data Transformation	Decide how to represent the data, sampling rate of data.	Don e	Data has been analyzed and cleaning of the has been performed
3	Create UI	Create UI to access the data stream pipeline.	Done	Create a UI to select and understand the technical, fundamental and insight of the selected stock.
4	Data Analysis & EDA	Analysis of data and doing EDA	Done	EDA of data has been performed Based on that we have selected the important features
5	Feature Extraction	Feature Extraction	Done	EDA of data is in progress. Based on that we will select the important features
6	Create & train suitable	Create & train suitable model	Done	We have tried different models and checked the accuracy of each model.
7	Model Evaluation	Evaluate the models and choose the effective model.	Done	Have tried various Linear regression, K means clustering, RNN models and fbprophet model for time series data..
8	Document the results and prepare final report	Document the results and prepare final report	Done	

4. Methodology

In order to understand the stock market, we must understand the business, domain and current affairs of the company. So to understand these aspects, we have divided the problems into various small modules.

1. Graph based technical analysis.
2. Using the time series based future stock price prediction.
3. Cluster based stocks grouping.
4. Finding the top performing stocks based on historical data.
5. Linear regression models coefficient based correlations.
6. News based sentiment analysis.

1. Graph-based technical analysis.

Rolling statistics is a major tool in the so-called technical analysis of stocks, as compared to the fundamental analysis which focuses, for instance, on financial reports and the strategic positions of the company whose stock is being analyzed.

A decades-old trading strategy based on technical analysis is using two simple moving averages (SMAs). The idea is that the trader should go long on a stock (or financial instrument in general) when the shorter-term SMA is above the longer-term SMA and should go short when the opposite holds true. The concepts can be made precise with pandas and displaying the data on the spread chart

2. Using the time series-based future stock price prediction

We used historical data from yahoo's finance api's to predict the stock price for a given stock symbol, given other stock symbols. In addition to the normal pandas and numpy libraries, We are also using Prophet, which is a forecasting library released by the Facebook core data science team. It implements a procedure for forecasting trends that fit with yearly, weekly, and daily seasonality, plus holiday effects.This gives us the future trend predictions for the stock provided as input.

In summary, based on the three types of data obtained from the methods above, we will be able to predict specific results out of data. Also, we will build a comprehensive recommendation system out of this data to investors.

3. **Cluster based stocks grouping**

We have collected the stocks fundamental data like price, P/E ratio, Volume, market capitalization and EPS. Based on this data, we have built a vector model to group the similar stocks based on the features selected by the user using K-Mean cluster technique. We have also added a variation by adding the level of feature based similar stocks by changing the cluster size.

4. **Finding the top performing stocks & Linear regression models coefficient based correlations**

We build a simple linear regression model for each symbol and calculate the coefficients representing trends. As what we do in class, we use the linear model. LinearRegression function from scikit-learn library. To run the model, first divide the historical stock price data into training and testing sets, then fit the model on the training set and predict with the fitted model on the testing set.

For fetching historical data from yahoo finance, package pandas_datareader is used. Since the linear regression model only applies for short time analysis, we only chose the date range from 1/1/2018 till latest data available, to get the latest trends.

Using popular symbol lists filtered from article content, we record coefficient value for every stock. Finally, we could output the top ‘n’ stocks with best trends to recommend to users. Also based on the coefficient of the selected stock, we can suggest the similar stocks, whose coefficients are nearby to the selected stock.

5. **News based sentiment analysis**

Using googlenews python api, we are downloading all the news for the last 5 days and using *SentimentIntensityAnalyzer* python library, we are calculating the sentimental score and bifurcating the score into positive, neutral and negative sentiment score, based on the polarity scores of news.

5. Code & Results

All code used for this project has been committed to the git link below.

- https://github.com/arungupta26/stock-recomendation-system

1. **Using the time series-based future stock price prediction**

We have used the fbprophet library for building the models for all the stocks mentioned in the stock list file.The data is then normalized (take log of all closing values to normalize the skewed data) and we can create a Prophet model with seasonality and can make a forecast prediction which allows an investor to track overall behavior of stock investments and decide if the model is no longer applicable if the value at any point is outside the confidence interval bounds.

We have used the last 4 year data to create the model. Once models are created, we are saving the model to the resources folder with name format <STOCK_NAME>.model, so that once we want to predict the future stock price then based on the stock name we can select the model and create a model instance using pickle api.

Below is the screenshot(Fig 1) for downloading the historic data and modeling the model to save in the resources folder.

```
# Considering data from 2000 to 2022
start = datetime.datetime(2018, 1, 1)
end = datetime.datetime(2022, 1, 22)

stocks = Path('./../../resources/stock_list.txt').read_text().split("\n")
index=1
for s in stocks:
    print(s)
    df = yf.download( s, start , end)
    close = df['Close']

    close_df = close.reset_index().rename(columns={'Date':'ds', 'Close':'y'})
    close_df['y'] = np.log(close_df['y'])
    model = Prophet(daily_seasonality=True)
    model.fit(close_df)
    pkl_path = "./../../resources/fbprophet/model/"+s+".model"
    with open(pkl_path, "wb") as f:
        pickle.dump(model, f)

#     # save the dataframe
#     forecast.to_pickle("./../../resources/fbprophet/forecast/"+s+".forecast")
    print(index,"*** Data Saved for -> ",s)
    index = index+1
```

Fig 1

Once we save the model, and the user selects the stock on UI, this model will be used to predict the future prices for next 10 days. Below screenshot(Fig 2) has been used to create the in memory model instance and can be used to predict the future prices.

```
with open('./../../resources/fbprophet/model/'+ selected_stock+'.pkl', 'rb') as f:
    m = pickle.load(f)
future = m.make_future_dataframe(periods=10)
forecast = m.predict(future)
```

Fig 2

This forecast instance has information of predicted prices of the next days, which will be displayed on the UI. Below is the screenshot(Fig 3) of the UI. Predicted Price has two more predictions such as predicted upper price and predicted lower price.

Last day closing price for TATAMOTORS.NS was Rs:490.5500

Future dates	Predicted Price	Predicted Price(Upper)	Predicted Price(Lower)
01/25/2022	545.5505	592.8934	496.2323
01/26/2022	544.7642	595.3940	498.4206
01/27/2022	543.6835	591.8821	497.8476
01/28/2022	539.3829	592.8570	491.5766
01/29/2022	507.3837	554.5375	464.1658
01/30/2022	532.4504	577.8558	482.9190
01/31/2022	535.4881	589.2919	490.8036
02/01/2022	534.1605	586.4026	490.6871
02/02/2022	533.1617	580.9690	486.1620
02/03/2022	532.2091	580.8384	481.8475

Fig 3

2. Cluster based stocks grouping

_____Here we have used clustering techniques based on the user selected features from sklearn.cluster to classify different stocks. Features available for the user as are below.

- Price
- Volume
- Market Cap
- Beta
- PE Ratio
- EPS

Different users are interested in different stock features. In order to facilitate the users and we make our recommendation more flexible, here we allow users to specify stock features they are interested in. They can select one or multiple features from 6 as shown above. The core codes of predicting the cluster each stock belongs to is below screenshot (Fig 4),

```
kmeans = KMeans(n_clusters=clusterNum,init='k-means++',max_iter=300,n_init=10,random_state=0)
clusterIds = kmeans.fit_predict(features[columns])
```

Fig 4

Based on the cluster allocated to the selected stock, we display all the others stocks in the same cluster as interested stocks. Below is the UI screenshot(Fig 5)

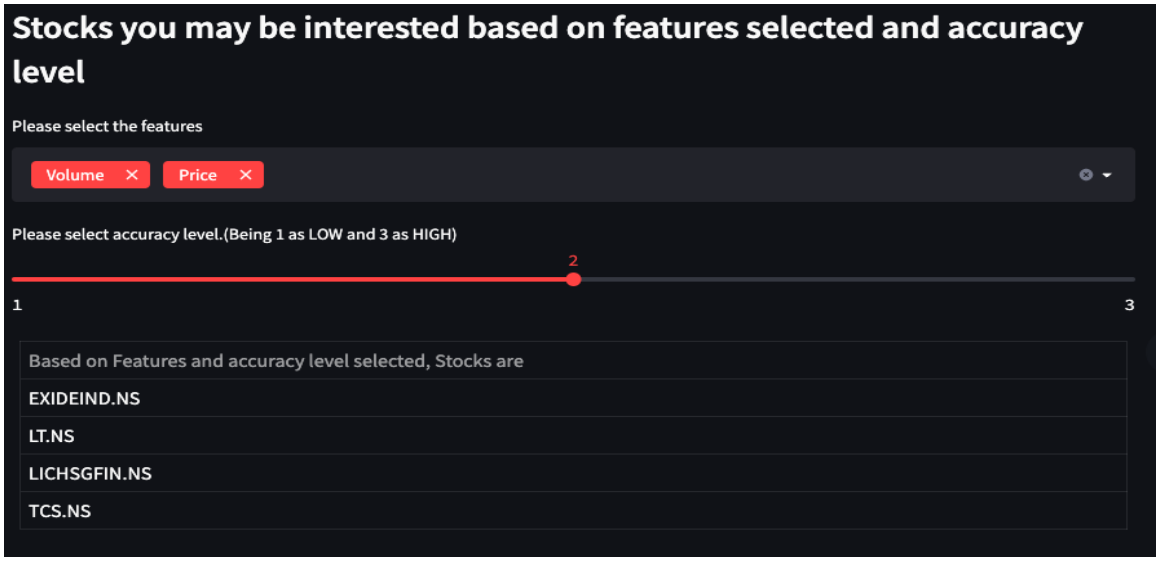


Fig 5

3. **<u>Finding the top performing stocks & Linear regression models coefficient based correlations</u>**

We have downloaded every stock (present in stock list file,around 170 stocks) data from yahoo finance API. As per the data downloaded, we have built a linear regression model for each stock to get the coefficients representing the upward or downward trend of the stock over the span of 4 years. Sample historic data of one of the stocks is attached in the below screenshot(Fig 6).

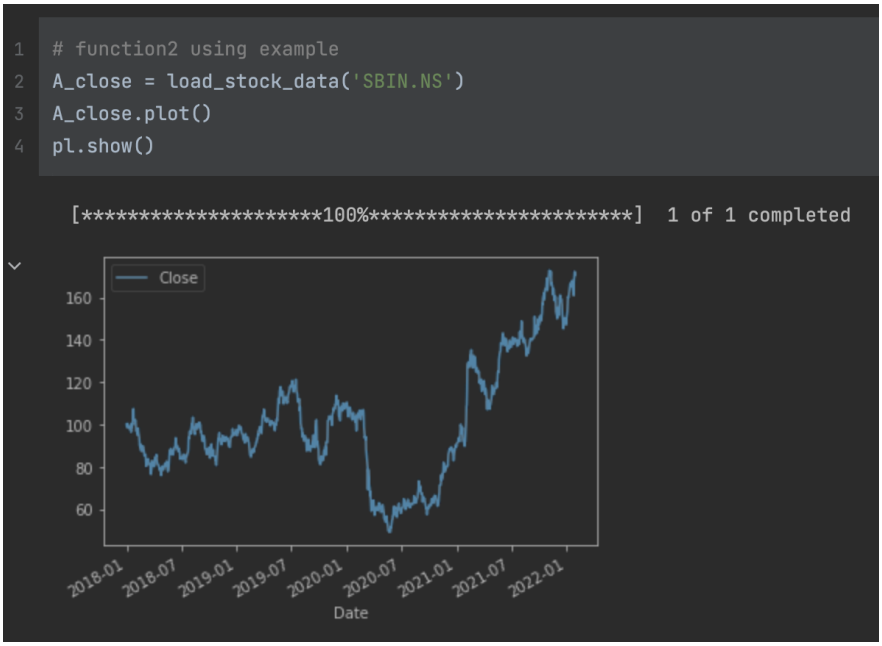


Fig 6

Then we train the linear regression model specific stock. From the trained model, we get the coefficient value and save it to a csv file for all the stocks, so that those values can be used for single day recommendation, as these values will be constant at least for a day. The linear regression model is evaluated by mean square and variance score. Variance score is the degrees to which the model explains the variations present in the data.

These scores are used to measure the efficiency of a particular linear model. Variance value to 1 means the perfect prediction, so we could use these values to analyze the performance of the model.

Sample stock with linear model prediction is attached below(fig 7),

```
build_linear_regression('SBIN.NS')

[**********************100%***********************]  1 of 1 completed
Coefficients          -> [3.06566898]
Mean squared error    -> 598.72
Variance score        -> 0.29
features              -> Date Stock Close Price

3.065668979179467
```

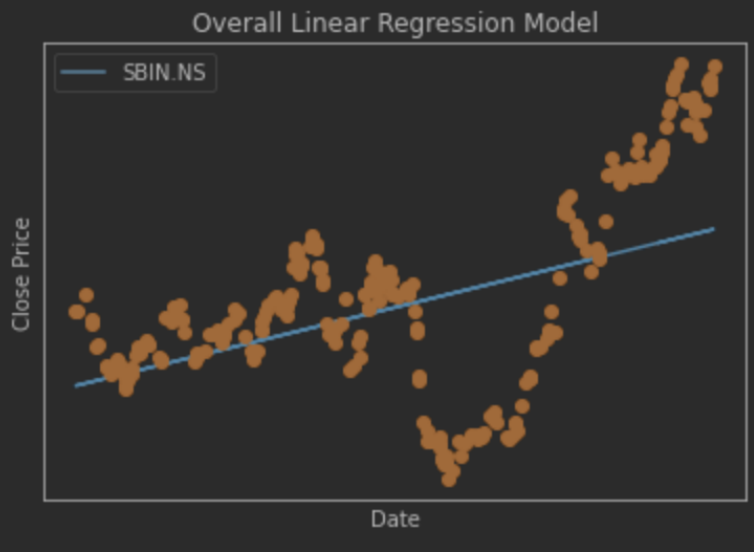


Fig 7

Based on these coefficient values, stocks with the highest positive score will have the greatest linear slope, which means that this stock has the highest growing pattern among all the other stocks. This is how we calculate the top performing & recommended stock to invest to the user. This is what we recommend to the user on UI (Fig 8).

Fig 8.1

```
# function5 using example with coefficient_data_test
get_top_stock(coefficient_data_test, n = 5, show_dots = True)
```

```
[**********************100%**********************]  1 of 1 completed
[**********************100%**********************]  1 of 1 completed
[**********************100%**********************]  1 of 1 completed
[**********************100%**********************]  1 of 1 completed
[**********************100%**********************]  1 of 1 completed
The 5 stocks with best trends are:
['ADANIENT.NS', 'INDIAMART.NS', 'DEEPAKNTR.NS', 'IRCTC.NS', 'COFORGE.NS']
```

	Stock	Coefficient
3	ADANIENT.NS	113.13193
84	INDIAMART.NS	66.54931
49	DEEPAKNTR.NS	64.641476
87	IRCTC.NS	54.489549
41	COFORGE.NS	43.986951

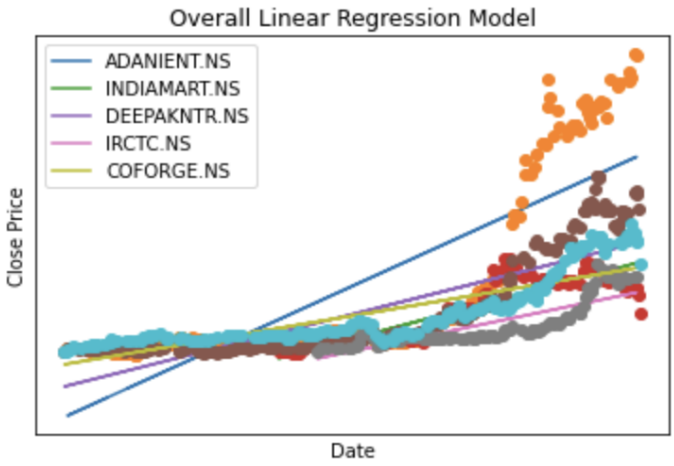
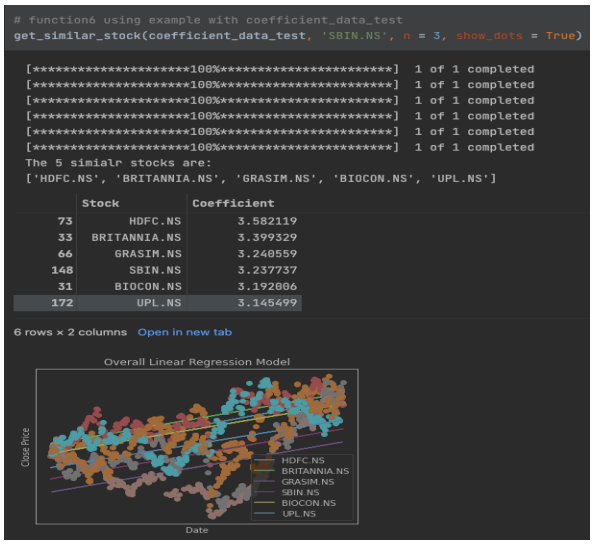


Fig 8.2

Again, based on the same coefficient values, we can get ‘n’ similar stocks, as per the stock selected by the user on UI. We achieve this by getting the offset of +/- n/2 stock around the coefficient of the selected stock. Same stocks are displayed on the User UI. Shown in below screenshots.



```
# function6 using example with coefficient_data_test
get_similar_stock(coefficient_data_test, 'SBIN.NS', n = 3, show_dots = True)

[**********100%**********]  1 of 1 completed
[**********100%**********]  1 of 1 completed
[**********100%**********]  1 of 1 completed
[**********100%**********]  1 of 1 completed
[**********100%**********]  1 of 1 completed
[**********100%**********]  1 of 1 completed
The 5 simialr stocks are:
['HDFC.NS', 'BRITANNIA.NS', 'GRASIM.NS', 'BIOCON.NS', 'UPL.NS']
```

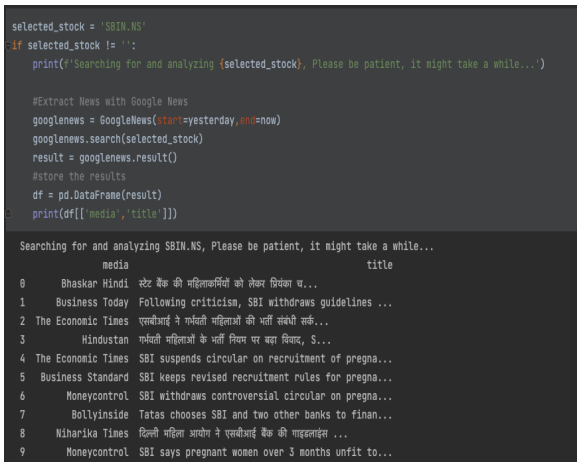
	Stock	Coefficient
73	HDFC.NS	3.582119
33	BRITANNIA.NS	3.399329
66	GRASIM.NS	3.240559
148	SBIN.NS	3.237737
31	BIOCON.NS	3.192006
172	UPL.NS	3.145499

6 rows × 2 columns Open in new tab

Fig 9

4. News based sentiment analysis

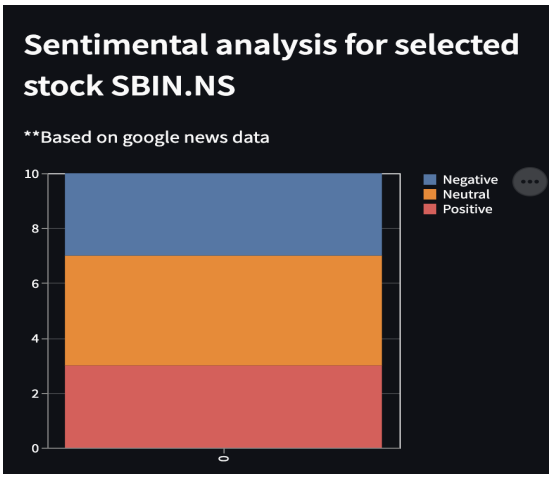
News plays an important role to decide whether to buy/sell a stock. Because the trading of stocks also depends on the sentiment of the market as well. We have done the sentiment analysis of the news of the selected stocks. We have used the Google news python based library api to get the news around the selected stocks. Based on the summary of the stock news, we are calculating the polarity value using the SentimentIntensityAnalyzer Python library. Below screenshot have news and sentimental calculating codes.(Fig 10)



```
selected_stock = 'SBIN.NS'
if selected_stock != '':
    print(f'Searching for and analyzing {selected_stock}, Please be patient, it might take a while...')

    #Extract News with Google News
    googlenews = GoogleNews(start=yesterday,end=now)
    googlenews.search(selected_stock)
    result = googlenews.result()
    #store the results
    df = pd.DataFrame(result)
    print(df[['media','title']])

Searching for and analyzing SBIN.NS, Please be patient, it might take a while...
                media                                              title
0        Bhaskar Hindi  स्टेट बैंक की महिलाकर्मियों को लेकर प्रियंका च...
1       Business Today  Following criticism, SBI withdraws guidelines ...
2   The Economic Times  एसबीआई ने गर्भवती महिलाओं की भर्ती संबंधी सर्क...
3           Hindustan  गर्भवती महिलाओं के भर्ती नियम पर बढ़ा विवाद, S...
4   The Economic Times  SBI suspends circular on recruitment of pregna...
5   Business Standard  SBI keeps revised recruitment rules for pregna...
6        Moneycontrol  SBI withdraws controversial circular on pregna...
7         Bollyinside  Tatas chooses SBI and two other banks to finan...
8      Niharika Times  दिल्ली महिला आयोग ने एसबीआई बैंक की गाइडलाइंस ...
9        Moneycontrol  SBI says pregnant women over 3 months unfit to...
```



6. Technologies Used

a. **Python libraries**
    i. Numpy, Pandas, seaborn, matplotlib sklearn, plotly, fbprophet
b. **Streamlit**
    i. Used for front-end and plotting the various useful visualizations on user interface.
c. **Server**
    i. Aws ec2 instance for deploying applications.

7. Challenges

a. We tried to integrate the twitter sentiment analysis, but due to unavailability of the access token and access key by twitter, we were unable to do it.
b. While trying the other algorithms to train and test the models to predict the future stock price, the predictions were not accurate.

8. Future work

a. Currently, single stock based analysis is supported. Later multiple stocks should be analyzed simultaneously.
b. UI performance can be improved with easy/attractive and faster loading time.
c. Models used to predict the future stock price can be trained with different algorithms as well, to test and predict the better/close predictions.
d. News sentiment analysis can be integrated with other sources as well.
e. Streaming of stock data can be implemented so that recommendation will be close to real time.
f. Model refreshment can be done in a regular & short interval of time to avoid any inaccuracy.

9. Conclusion & Summary

Overall, we feel that these four effective detailed analyses do provide a method for users to identify healthy stock options as well as groups of stock similar to a specific stock symbol as shown by each of the individual results in each category.

There were two basic philosophical approaches we took to help solve the issue of aiding investors in stock purchases. One was that past behavior of stock prices is an indicator of future behavior (data modeling)

and the second was that the best way to predict tomorrow's stock value is to look at today's price as well as the price of the similar stock (associative modeling). We feel that both of these do help guide investors.

The hard part of making predictions in such a volatile environment is that there are so many factors that affect the stock price on a day-to-day basis. We tried to correct some of the more obvious ones like seasonality and some amount of volatility and unforeseen events by taking both data modeling techniques along with associative modeling techniques.

Our results showed that it is possible to correlate different stocks based on factors like linear regression modeling coefficients as well as being able to make models forecast future performance, and this data can all be used to give guidance to potential investors. However, it's difficult to tell which stock to invest in.

The conclusion that we can make at this point is that our system gives guidance, however, we feel that a more complete solution would be the aggregation of these methods into a system that provides a single set of investment opportunities. The selection of possible stock investments is still heavily relied upon by the user to make.

10. Reference

a. https://www.kaggle.com
b. https://finance.yahoo.com
c. https://pythonforfinance.net/2018/02/08/stock-clusters-using-k-means-algorithm-in-python/
d. https://github.com/NUOEL/cs6220
e. https://facebook.github.io/prophet/docs/quick_start.html
f. https://en.wikipedia.org/wiki/Stock_market_prediction