Software Engineering of Web Applications (16:332:568)

Spring 2019

INVESTGENIE

A WEB APPLICATION FOR STOCK MARKET PREDICTION

TEAM 2

Ashwin Channakeshava

Kartik Rattan

Mareesh Kumar Issar

Suraj Nagaraja

Sarthak Vengurlekar

SUBMISSION DATE: 05/09/2019

TABLE OF CONTENTS

1. CONTRIBUTION BREAKDOWN

All group members contributed equally. The team members assisted each other in developing all the modules.

1.1 Customer Statement of Requirements

Buying stocks refers to obtaining holdings of a company. Stocks are categorized as a part of company's possession. When individuals purchase a stock, they invest in the stakes of a company's profit. Companies sell their stocks in order to gain capital to fund their expansion. The growth of a company is directly proportional to the amount of stocks bought by individuals. Buying stocks has its own risks and rewards. If a company goes into loss, the worth of stocks of that company decrease. This results in loss of money invested. Understanding the risks and rewards of investing stock market is very crucial in stock market exchange.

1.2 Motivation

From its birth, the stock market has always been a place where businessmen could gain capital buy selling shares of their companies and where investors can purchase these shares in hope to make a profit when those companies prosper. Although the market still serves both these purposes, today it is judged less by what it does for businessmen seeking capital but rather for what it can do for investors seeking gain. In a time of high taxes and low wages, trading stocks seems like a viable and almost effortless way of becoming very rich very quickly. Although this scenario has proven true for many, there are just as many, if not more, who can proclaim the exact opposite. The problem here lies with the fact that the same volatility in the market that can be exploited for gain, can be the cause for loss as well.

Computers now days are capable of computing various operations. With Internet in the picture, it is possible to browse stocks right from individuals own PCs. Buying, selling and managing stocks is become very simple, affordable and convenient to people. In the past, contact with trained professionals was necessary for stock market trading. For casual investors who are looking to make long term or short term profits trading stocks, they need to know the risks precedent to buying particular stocks. A tool is required to compare the stocks, know the past prices of a particular stock and the gains and losses that were associated with a particular stock. Most people do not have the time and the knowledge to differentiate between potential stocks to buy.

1.3 Vision

The computer can provide information of the stocks in a timely manner and stock related information. Operating a computer is easy and almost all working professionals have the knowledge of running a computer or a phone.

So far, the use of software to aid those of us looking for quick and effortless stock forecasts has been conceptually realized, but what about those that not only want a decision but also aspire to learn the process behind the decision? Some might argue that this will defeat the purpose of not having to be knowledgeable to get reasonable stock predictions. The beauty of this feature, however, is that it is completely optional and thus will serve both sides of the crowd: those that just want quick, easy, and well tested advice to base their decisions on and those that are looking for a little more reasoning behind those decisions. Thus, differing from other designs, each prediction result should have a way to reveal an explanation to the user of the technical analysis used to get to that specific result; an environment for users to learn similar processes used by professionals if he/she chooses to.

The analysis of the stocks is based on the behavior of the stocks in the past and the present. The historical data of the stock is very important when it comes to predicting the future state of the stock. Using various tools and technical indicators, prediction of the future prices of the stocks is undemanding and precise to a certain extent. Using this, investors can understand the basics of stock market prediction and the risk subjected to the ownership of particular stocks.

Having all these factors in mind, we have developed a web-based stock market system which helps the users to obtain the price of stocks. This system also predicts the future variations in the price of the stocks for the user to determine if the stock is worth the funds. The system comprises of three basic levels. The first level being the front-end. The User Interface helps the user to browse through the web page easily and get information about stocks on their screen. The second level is the back-end where the data collection takes place. The implementation of technical indicators is also done back-end. The third level is the implementation of machine learning using Artificial Neural Networks, Bayesian Curve Fitting and SVM. Selecting a stock on the web page will display information about the stock and the option to use various indicators on the stock for prediction. Prediction is done at the back-end to predict the future prices of the stocks.

2. GLOSSARY OF TERMS

1. Stock Market: A Stock exchange.
2. Prediction Model: Predictive modeling is a process that uses data mining and probability to forecast outcomes**.** Every model has predictors that help in determining the future results.
3. Algorithms: A set of rules to be followed in calculations or other problems to be solved.
4. Machine Learning: An application of Artificial Intelligence is Machine Learning. It can learn and predict outcomes without being explicitly programmed.

5. Neural Network: It is based on human central nervous system.
6. Opening Price: It is the price of a stock at the opening exchange of a trading day.
7. Closing Price: It is the price of a stock at the end of the day's business. It is the most up to date value of a stock until the next day.
8. Charts: The chart consists of various vertical lines representing price range and horizontal line representing the opening and closing price.
9. Database: A well-arranged collection of data regulated to support being processed for various functions.
10. HTML: HTML stands for Hypertext Markup Language. It is the standard markup language that is used for creating web pages.
11. CSS: CSS stands for Cascading Style Sheets. It is the language that is used to specify and modify the design of web pages.
12. CDN: CDN stands for Content Delivery Network. It is a network of geographically distributed servers that facilitate fast and reliable data delivery of content to the user.
13. URI: URI stands for Uniform Resource Identifier. It refers to a string of characters that uniquely identify a resource.

3. SYSTEM REQUIREMENTS

3.1 Enumerated functional requirements

IDENTIFIER	PRIORITY	REQUIREMENTS
REQ1	**4**	The system will allow users to register by creating a username and setting a password
REQ2	**4**	The system will allow users to access the main page by entering valid username and password.

REQ3	**5**	The system will allow the user after login to obtain graphical representation of the stock selected by the user
REQ4	**5**	The system will allow the user to choose from five queries to be applied on the stock of any company.
REQ5	**4**	The system will allow users to select any option from the given stock list
REQ6	**5**	The system will apply different prediction models and use indicators to give a summarized result on whether to sell, buy or hold a stock.
REQ7	**4**	The system will allow the users to publish his or her thoughts in the form of a comment with their name.
REQ8	**4**	The system will direct the user when they publish a comment to a blog page which has all comments posted by them and others as well (in chronological order).
REQ9	**5**	The system will pull the information of the stocks which has been stored in the database as requested by the user
REQ10	**5**	The system will store both historical and current stock prices in the database for all the stocks in the stock list

3.2 ON-SCREEN APPEARANCE REQUIREMENTS

IDENTIFIER	PRIORITY	REQUIREMENTS
REQ1	**5**	A drop-down menu for the user to navigate through the webpage. One drop-down menu allows the user to select the stock to be viewed. Queries can be selected with the help of the second drop-down menu. A third drop-down menu helps user in selecting the graph time period.
REQ2	**5**	To maintain proper usability, consistent design and good functionality, the page must adapt to various sizes depending upon the size of the window and the type of device used.
REQ3	**4**	The web application must support various browsers such as Microsoft Edge, Google Chrome and Mozilla Firefox all updated to their latest versions.
REQ4	**4**	The UI must be clear and easy to understand. The user should be able to navigate through the page easily. The wordings must be clear and direct.

4. FUNCTIONAL REQUIREMENTS SPECIFICATION

4.1 Stakeholders

There are two Stakeholders in this configuration:

1. Administrator: Manages user accounts. Adds/removes and updates the stocks.
2. User: Can create or delete an account. Any user can log in to the system to check stock predictions.

4.2 Actors and Goals

There are two actors in our web application system. The user and the administrator.

User: An ordinary user of the web application.

Administrator: The handler who oversees updating the system periodically and keeping it in working order.

Database: All the information of the stocks is stored in the database. The user information is also stored in the database.

Prediction Algorithm: Algorithms to calculate the prediction. For example, Bayesian Regression, Artificial Neural Networks, and Support Vector Regression.

Graph: Provide charts, that plot the data. The plots are time vs price.

Alpha Vantage API: This is the API from where the real time and historic stock prices will be obtained from.

4.3 Use Cases

USE CASE	USER	FUNCTION
Case 1	Administrator	Manage the application
Case 2	User	Login
Case 3	User	Register
Case 4	User	Display Graph
Case 5	User	Select Stock
Case 6	User	Stock Prediction
Case 7	User	Comments
Case 8	User	Display Stock Price
Case 9	User	Recommendation

Use Case 1: Manage the Application

The administrator can add/remove stocks and manage user accounts. The administrator can also create the database and update the database.

Use Case 2: Login

Input the user ID and password to login to the application.

Use Case 3: Register

Register with a user ID and password to use the application.

Use Case 4: Display Graph

Displays the graph for historical/real-time data for the selected stock.

Use Case 5: Select Stock

Select a particular stock from a list of predefined stock list.

Use Case 6: Stock Prediction

The application predicts the future value of the stock depending on the previous values of the stock prices.

Use Case 7: Comment

The user has an option of leaving his/her thoughts on a particular stock in a comments box provided below each stock.

Use Case 8: Display Stock Price

Display the stock information including open, high, low, close, and volume.

Use Case 9: Recommendation

The application advises the user whether it is a good time to buy, sell or hold on a stock.

5. EFFORT ESTIMATION

5.1 Unadjusted Actor Weight

ACTOR NAME	DESCRIPTION	COMPLEXITY	WEIGHT
User	The user uses the system to get stock information.	Average	2
Administrator	The Administrator oversees the system and manages the stock related functions and users.	Complex	3
Database	All the stock information and user information along with comments is stored in the database.	Complex	3
Prediction Algorithm	Algorithms to predict the future behavior of the stock.	Complex	3
Graph	Provides charts that plot the data.	Average	2
API	API provides the results	Complex	3

Total UAW = 2 + 3 + 3 + 3 + 2 + 3 = 16

5.2 Unadjusted Use Case

USE CASE	COMPLEXITY	FUNCTION	WEIGHT
Case 1	Complex	Manage the application	9
Case 2	Average	Login	6
Case 3	Average	Register	6
Case 4	Complex	Display Graph	9
Case 5	Complex	Select Stock	9
Case 6	Complex	Stock Prediction	9
Case 7	Average	Comment	6
Case 8	Complex	Display Stock Price	9
Case 9	Complex	Recommendation	9

Total UUCW = 9 + 6 + 6 + 9 + 9 + 9 + 6 + 9 + 9 = 72

Total UUCP = UAW + UUCW = 16 + 72 = 88

6. DOMAIN ANALYSIS

6.1. Concept Definitions:

The domain concepts and their definitions are defined below.

Website:

Definition-

- A hypertext document which is connected to the web.

Responsibilities-

- Display content to the users via HTML pages which are styled using CSS.
- Allow users to interact with the website via appropriate buttons and menus.

Query:

Definition-

- Search query which performs a particular function.

Responsibilities-

- Hold a specific search query.

Predictor:

Definition-

- Predict the future value for the chosen stock.

Responsibilities:

- Apply prediction algorithms to historical data and to estimate future values.

Data Collector:

Definition-

- Module to fetch stock data.

Responsibilities-

- Send an API call to the data provider and fetch stock data, both real time and historical.
- Retains the stock data in database for further operations.

Timer:

Definition-

- Function to set the time between pull requests from the data provider.

Responsibilities-

- Update the fetched data based on the specified time interval.

Database:

Definition-

- A MySQL database with multiple tables for the real and historical prices of various stocks.

Responsibilities:

- Connect to the data collection module for updates and to provide stock data when required.
- Store various stock data in tables.

6.2 Attributes:

Website- The attributes for the website are related to display text, graphs and comments.

Query- Holds attributes related to search queries for various stocks, prices and time durations.

Database- Hold attributes to connect to the local MySQL database including address, port number, password and user name.

Timer- Holds attributes related to when to update stock prices and the duration of future prediction.

7. CLASS DIAGRAM AND INTERFACE SPECIFICATION

7.1 Class Diagram:

Overview

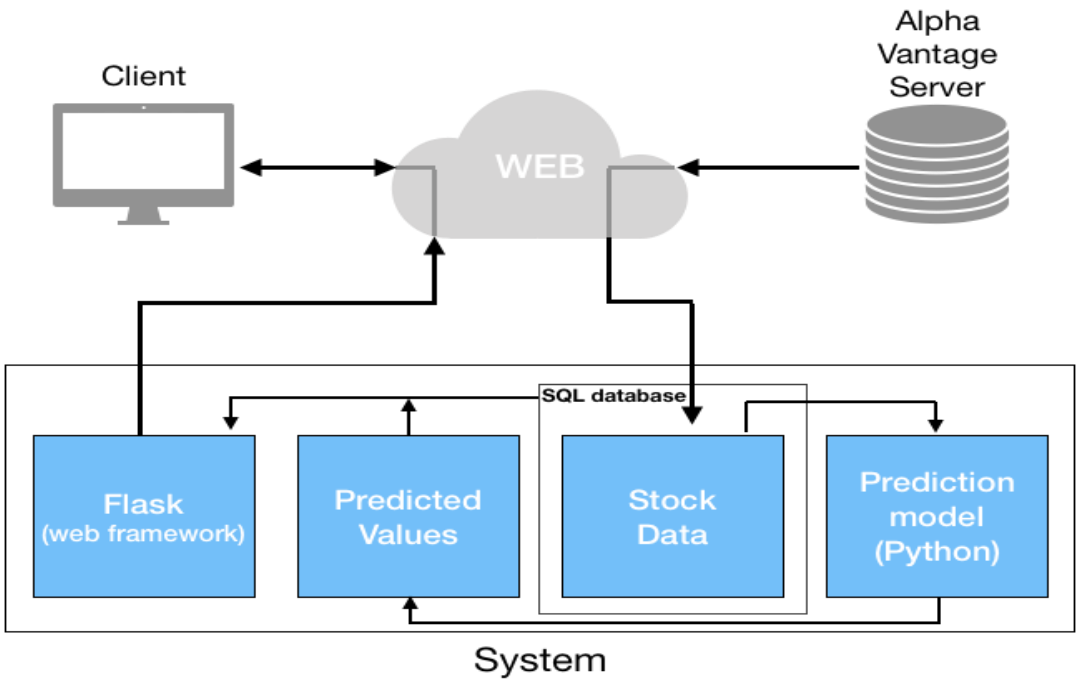


Fig. System Architecture.

Time sequence diagram:

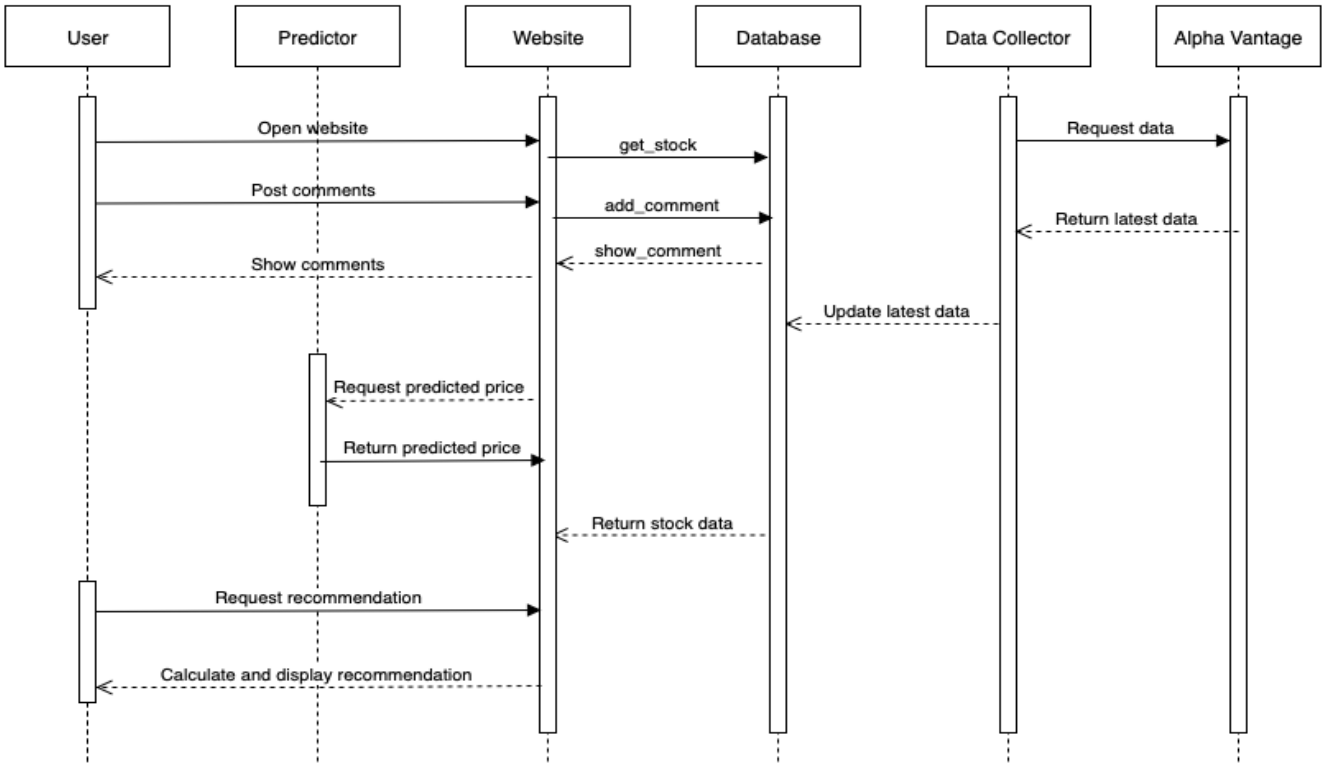


Fig. Timing sequence diagram.

7.2 Function signatures:

- show_open_close(String stock_symbol):array

  Returns the open, high, low, close and volume values of the chosen stock.

- prediction(String stock_symbol):String

  Returns the recommendation - BUY, SELL or HOLD based on indicators.

- fig(String stock_symbol, String period):void

  Displays a graph of closing prices vs time for the selected stock symbol and period.

- pred_final(String stock_symbol):Float

  Returns the predicted future stock value using the average of the three prediction algorithms.

- queries(String stock_symbol, String query_type):array

  Returns the result for the selected predefined query.

- api_queries(String stock_symbol, String query_type):array

  Returns the result corresponding to selected API query.

8. SYSTEM ARCHITECTURE AND SYSTEM DESIGN

8.1 Architecture styles:

Bootstrap

Bootstrap is an open source HTML, CSS and JavaScript framework that allows users to develop the front-end of their web applications. It serves as a front-end component library, where the users can select from a wide variety of available templates for their web application. In order to use these templates, the user needs to include links to their CDN servers (where template is present) in their html document. These templates are complete with all the buttons, forms, navigation bar etc. After adding the template, the user needs to define the functionality only for the components that he/she requires for their application.

Another advantage that bootstrap provides over conventional webpage designing is faster and convenient A/B testing. A/B testing refers to the method of testing two different versions of a website, in order to determine which version performs better in terms of driving more traffic to the website (by showing one of the randomly selected version to the user and doing statistical analysis to find which version performs better).

8.2 Data Storage:

The real-time and historic data that is received through the alphavantage API is stored in MySQL database. MySQL is an open-source RDBMS that serves as the underlying database for most of the large internet companies. MySQL workbench is used as a visual tool for managing the databases.

The historic table contains the data for each of the predefined stock symbols that is updated at the end of each day.

The design of the historic table is as follows: -

COLUMN NAME	TYPE	REMARK
Date_Time	DATE	The date when stock was traded
Open	DOUBLE	Opening price of the stock for that day
High	DOUBLE	Highest price of the stock for that day
Low	DOUBLE	Lowest price of the stock for that day
Close	DOUBLE	Closing price of the stock for that day
Volume	DOUBLE	Volume of the stock traded for that day

The real-time table for all the stock symbols runs in the background to fetch the entries for that day in 1-minute intervals.

The design of the real time table is as follows: -

COLUMN NAME	TYPE	REMARK
Date_Time	DATE	The date when stock was traded
Close	DOUBLE	Closing price of the stock for that day for each 1 minute intervals
Volume	DOUBLE	Volume of the stock traded for that day for each 1 minute interval

The comments are stored in firebase database.

We have added a special feature where users can upload their thoughts in the form of comments so that others can view their opinions in the form of a blog. We use firebase database for this feature which is an online real-time database. There is a python library called python-firebase which allows us to connect our main python file to the database. Users are provided with a text box where they can fill out their comments. When the user presses the submit button, they are redirected to a blog page which has a list of the comments left by other users previously. Users can view all the comments irrespective of

the stocks on which the comments were made. Only users who submit a comment can access the blog page.

8.3 Web service:

SOAP- Simple Object Access Protocol and REST- Representational State Transfer are both webs service communication protocols. In recent years, REST is now representing most of the public APIs as compared to SOAP. While SOAP performs operations through a standardized set of messaging patterns (providing it with some inherent security features), REST directly access the data by using HTTP (Hypertext Transfer Protocol).

Following are some of the benefits of using REST over SOAP:-

1. REST supports wide variety of data formats like JSON (JavaScript Object Notation), XML etc. whereas SOAP supports only XML.
2. It is much more convenient to work with REST as compared to SOAP because the JSON format supported by REST can be easily parsed as compared to its XML counterpart,
3. It consumes less bandwidth and is easier to implement and understand for the developer and the user working with it.

REST

REST system is characterized by six design rules. Some of the important ones include separation between server that offers a service and client that consumes it, server cannot store the information given by the client for a request etc. Web resources in REST are represented by URIs. The client sends requests to these URI using HTTP protocol. Most common HTTP methods include GET, POST, PUT, and DELETE.

An API exposes the functionality of our application to the user.

We provide the user with JSON format output

For example,

Query- Get the highest stock price for any company in the last 10 days:

URL- localhost:5000/api2?stock_symbol=GOOG&query_type=high_val_ten

Result:

{
“date”: “Mon, 29 Apr 2019 00:00:00 GMT”,

“High”: 1287.58
}

Flask

Flask is a micro web framework (minimalistic web application framework) in python. A web application framework is a software framework that supports the development of web applications including web services, web resources and web APIs. Flask uses Jinja2 templating engine and Werkzeug, WSGI (Web Server Gateway Interface) web application library. It is used to host the HTML pages. Since, it is written is python, it helps us in integrating our prediction engine along with database using python. Using flask provides flexibility in design choice to the user as flask connects the back-end to the front-end but has no database abstraction layer. This allows the user the user to work with his/her familiar database. Another strong point in favor of using flask are the flask extensions (additional packages) that can add various application features like support for hashing passwords and sending email. Other features supported by flask are RESTful request dispatching, development server along with debugger and support for secure cookies (client-side sessions).

9. PREDICTION AND RECOMMENDATION

9.1 Support Vector Regression

In machine learning, support-vector machines are supervised learning models with associated learning algorithms that analyze data used for either classification or regression analysis.

We begin our discussion with classification which leads to regression. Given some data points each belonging to one of two (or more) classes, the goal is to decide which class a new data point will be in. There are many hyperplanes that might separate the two classes of data points. Our objective is to find a plane that has the maximum margin, i.e. the maximum distance between data points of both classes. This is called a linear classifier. One reasonable choice as the best hyperplane is the one that represents the largest separation, or margin, between the two classes. So, we choose the hyperplane so that the distance from it to the nearest data point on each side is maximized.

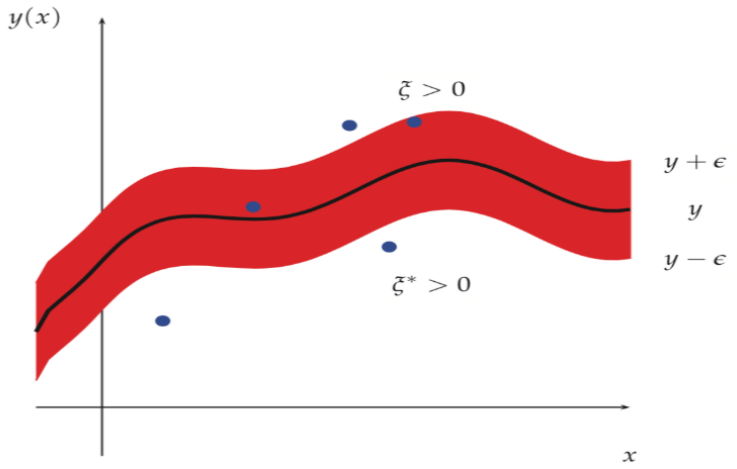


Fig. Support vector machines: maximum margin of error

In SVM, it is easy to have a linear hyper-plane between these two classes. But what if the data points are not linearly separable? In the XOR function, the data points cannot be separated linearly. To overcome this, SVM has a technique called the kernel trick. These are functions which takes low dimensional input space and transform it to a higher dimensional space i.e. it converts not separable problem to separable problem, these functions are called kernels.

Support Vector Machine can also be used as a regression method, maintaining all the main features that characterize the algorithm (maximal margin). The Support Vector Regression (SVR) uses the same principles as the SVM for classification, with only a few minor differences. Firstly, the output value is no longer binary (or n-ary for n classes). The output is a real number. This leads to increase in complexity as it becomes very difficult to predict the information at hand, which has infinite possibilities. In the case of regression, a margin of tolerance (epsilon) is set in approximation to the SVM. However, the main idea is always the same: to minimize error, individualizing the hyperplane which maximizes the margin, keeping in mind that part of the error is tolerated.

For our regression analysis, our input data set consisted of four features as close price, volume, high-low percentage change and open-close percentage change. The output label was closing prices 30 days into the future. Thus, the X [close, HLP, COP, volume] and the Y [close value 30 days into the future from X date]. We tested two kernels, rectified linear unit and linear model. We observed that the linear model gave us better accuracy and thus we implemented SVM algorithms using a linear kernel.

9.2 Artificial Neural Networks

Machine learning in simpler sense, was a technique implemented to make the machines learn and then predict future values. This began with providing a brain like toolset for the machines. McCulloch and Pitts Neuron (MP Neuron), in 1943 developed a computational mode inspired by the way biological neural networks in the human brain process information. It was a mathematical model of Neuron, consisting of weights w and threshold function with threshold Θ. Frank Rosenblatt invented the perceptron learning algorithm in 1957 and further increased the interest in the field. Unfortunately, for the growth this technique, Minsky and Papert (1969) showed that the without the hidden layers, the MLP was not able to classify data that is not linearly separable (for instance, XOR function).

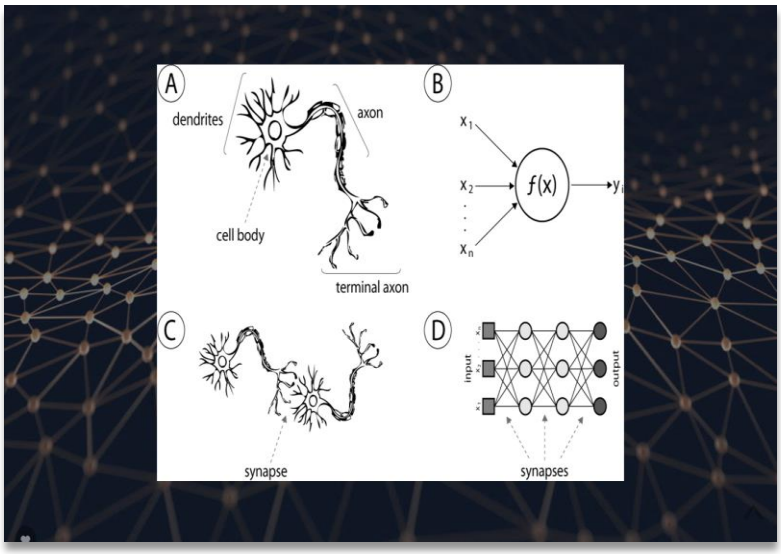
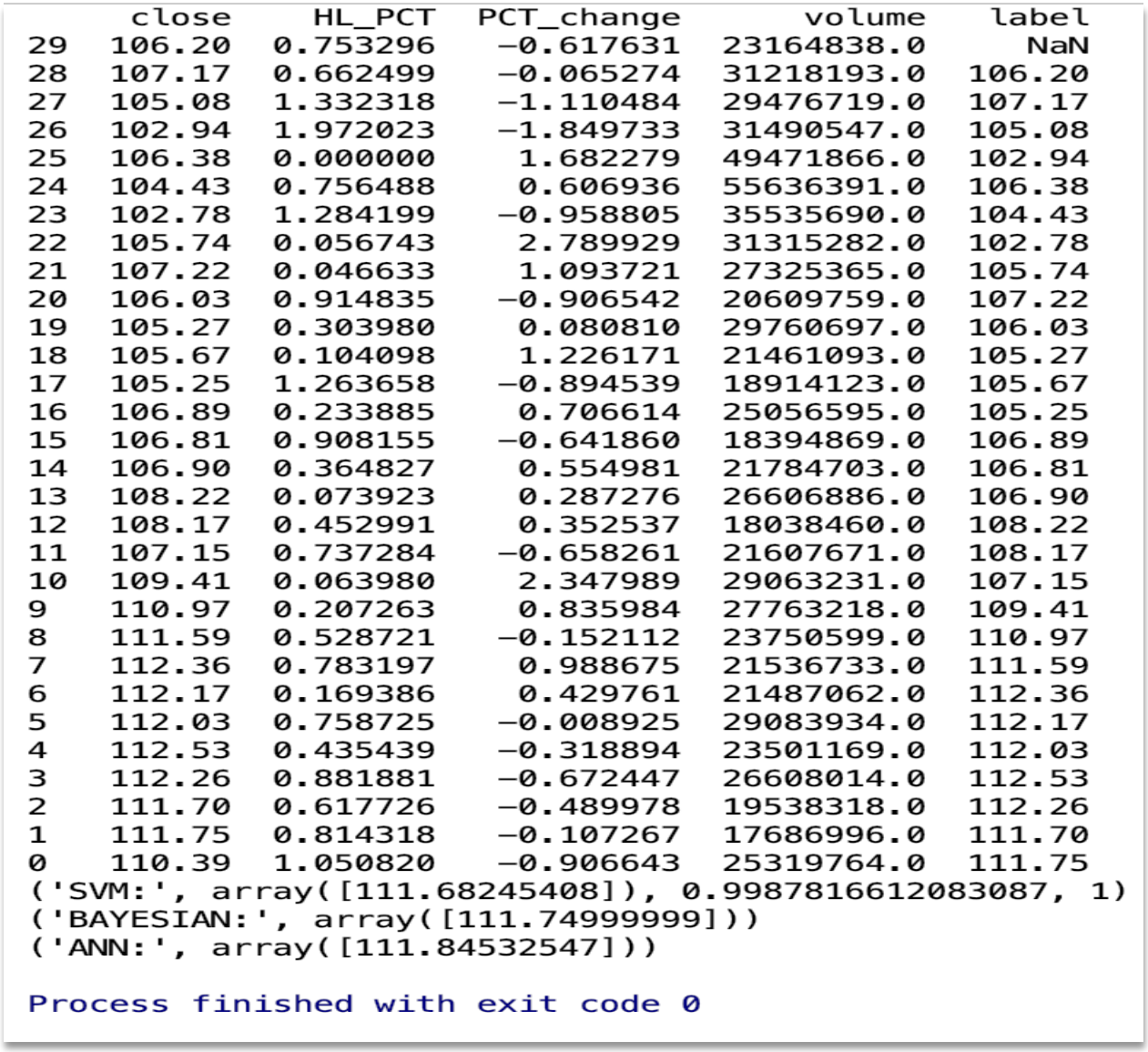


Fig. Mathematical model of MP neuron

We have an input vector: X [feature vector] with an output data point, y. ANN is a non-parametric machine learning technique since the number of parameters increases with an increase in number of inputs. We model the output at each perceptron with a sigmoid function, y_{out} = f(input) = 1 / 1 + e- (input). The output y from the neuron is computed with help of sigmoid function. The function f is non-linear and is called the Activation Function. The purpose of the activation function is to introduce non-linearity into the output of a neuron because most real-world data is nonlinear. Also, intuitively, the step function was not a very good estimator. For instance, a threshold value of 10 rejects an output value of 9.99. Thus, a smoother function was needed. In addition, it made the evaluations

simpler. Sigmoid function also has a value in between 0 and 1, and thus help us replace the binary output with probability. We define a learning rate ɳ. With a high learning rate, we can cover more ground each step, but we risk overshooting the lowest point since the slope of the hill is constantly changing. With a very low learning rate, we can confidently move in the direction of the negative gradient since we are recalculating it so frequently, but calculating the gradient is time-consuming. We then use a backpropagation algorithm to train our model. Backpropagation is our way to adjust weights. The method calculates the gradient of a loss function with respect to all the weights in the network. The gradient is fed to the optimization method which in turn uses it to update the weights, in an attempt to minimize the loss function.



	close	HL_PCT	PCT_change	volume	label
29	106.20	0.753296	−0.617631	23164838.0	NaN
28	107.17	0.662499	−0.065274	31218193.0	106.20
27	105.08	1.332318	−1.110484	29476719.0	107.17
26	102.94	1.972023	−1.849733	31490547.0	105.08
25	106.38	0.000000	1.682279	49471866.0	102.94
24	104.43	0.756488	0.606936	55636391.0	106.38
23	102.78	1.284199	−0.958805	35535690.0	104.43
22	105.74	0.056743	2.789929	31315282.0	102.78
21	107.22	0.046633	1.093721	27325365.0	105.74
20	106.03	0.914835	−0.906542	20609759.0	107.22
19	105.27	0.303980	0.080810	29760697.0	106.03
18	105.67	0.104098	1.226171	21461093.0	105.27
17	105.25	1.263658	−0.894539	18914123.0	105.67
16	106.89	0.233885	0.706614	25056595.0	105.25
15	106.81	0.908155	−0.641860	18394869.0	106.89
14	106.90	0.364827	0.554981	21784703.0	106.81
13	108.22	0.073923	0.287276	26606886.0	106.90
12	108.17	0.452991	0.352537	18038460.0	108.22
11	107.15	0.737284	−0.658261	21607671.0	108.17
10	109.41	0.063980	2.347989	29063231.0	107.15
9	110.97	0.207263	0.835984	27763218.0	109.41
8	111.59	0.528721	−0.152112	23750599.0	110.97
7	112.36	0.783197	0.988675	21536733.0	111.59
6	112.17	0.169386	0.429761	21487062.0	112.36
5	112.03	0.758725	−0.008925	29083934.0	112.17
4	112.53	0.435439	−0.318894	23501169.0	112.03
3	112.26	0.881881	−0.672447	26608014.0	112.53
2	111.70	0.617726	−0.489978	19538318.0	112.26
1	111.75	0.814318	−0.107267	17686996.0	111.70
0	110.39	1.050820	−0.906643	25319764.0	111.75

```
('SVM:', array([111.68245408]), 0.9987816612083087, 1)
('BAYESIAN:', array([111.74999999]))
('ANN:', array([111.84532547]))

Process finished with exit code 0
```

Fig. Output snapshot of our model

For our regression analysis, our input data set consisted of four features as close price, volume, high-low percentage change and open-close percentage change. The output label was closing prices 30 days into the future. Thus, the X [close, HLP, COP, volume] and the Y [close value 30 days into the future from X date]. MLP Regressor trains iteratively since at each time step the partial derivatives of the error function with respect to the model parameters are computed to update the parameters. We use the logistic sigmoid function as the activation function, which returns f(x) = 1 / (1 + e-x). The learning rate is defined as Ƞ = 0.01. We have a total of 4 layers. We set the first and the last layer input and output layer respectively. The remaining 2 layers are the hidden layer with 20 neurons units in each layer.

9.3 Bayesian Regression

The Bayesian viewpoint, we formulate linear regression using probability distributions rather than point estimates. The response, y, is not estimated as a single value, but is assumed to be drawn from a probability distribution. The model for Bayesian Linear Regression with the response sampled from a normal distribution is:

$$y \sim N(\beta^T X, \sigma^2 I)$$

The output, y is generated from a normal (Gaussian) Distribution characterized by a mean and variance. The mean for linear regression is the transpose of the weight matrix multiplied by the predictor matrix. The variance is the square of the standard deviation σ (multiplied by the Identity matrix because this is a multi-dimensional formulation of the model).

In the curve fitting problem, we are given the training data x and t, along with a new test point x, and our goal is to predict the value of t. We therefore wish to evaluate the predictive distribution p(t | x,x,t). Here we shall assume that the parameters α and β are fixed and known in advance. A Bayesian treatment simply corresponds to a consistent application of the sum and product rules of probability, which allow the predictive distribution to be written in the form:

$$p(t|x, \mathbf{x}, \mathbf{t}) = \int p(t|x, \mathbf{w}) p(\mathbf{w}|\mathbf{x}, \mathbf{t}) \, \mathrm{d}\mathbf{w}. \qquad (1.68)$$

p(t|x,w) in RHS: is given by (1.60), and we have omitted the dependence on α and β to simplify the notation.

p(w|x,t) in RHS: is the posterior distribution over parameters.

LHS: the integration in (1.68) can also be performed analytically with the result that the predictive distribution is given by a Gaussian of the form

$$p(t|x, \mathbf{x}, \mathbf{t}) = \mathcal{N}\left(t|m(x), s^2(x)\right) \tag{1.69}$$

where the mean and variance are given by

$$m(x) = \beta \boldsymbol{\phi}(x)^{\mathrm{T}} \mathbf{S} \sum_{n=1}^{N} \boldsymbol{\phi}(x_n) t_n \tag{1.70}$$

$$s^2(x) = \beta^{-1} + \boldsymbol{\phi}(x)^{\mathrm{T}} \mathbf{S} \boldsymbol{\phi}(x). \tag{1.71}$$

Here the matrix S is given by

$$\mathbf{S}^{-1} = \alpha \mathbf{I} + \beta \sum_{n=1}^{N} \boldsymbol{\phi}(x_n) \boldsymbol{\phi}(x)^{\mathrm{T}} \tag{1.72}$$

where I is the unit matrix, and the vector φ(x)=(1,x,x2,...,xM)T.

We predict the value for a stock using each of the above strategy. We then take an average of the three methods to return a final predicted value.

9.4 Recommendation

Technical Indicators:

1. Exponential Moving Average (EMA)

Exponential moving average is type of averaging that gives more weightage to the recent price data when compared to the older prices. The advantage of this is that it responds better to recent price changes. EMA is calculated the following way:

$$\mathbf{EMA=Price(t)\times k + EMA(y)\times(1-k)}$$

where:

t = today

y = yesterday

N = number of days in EMA

K = 2÷(N+1)

The Exponential Moving Average is used to produce buy and sell signals based on the crossovers and divergence from the historical average price data. Usually, 20-day, 30-day, 90-day and 200-day moving averages are used.

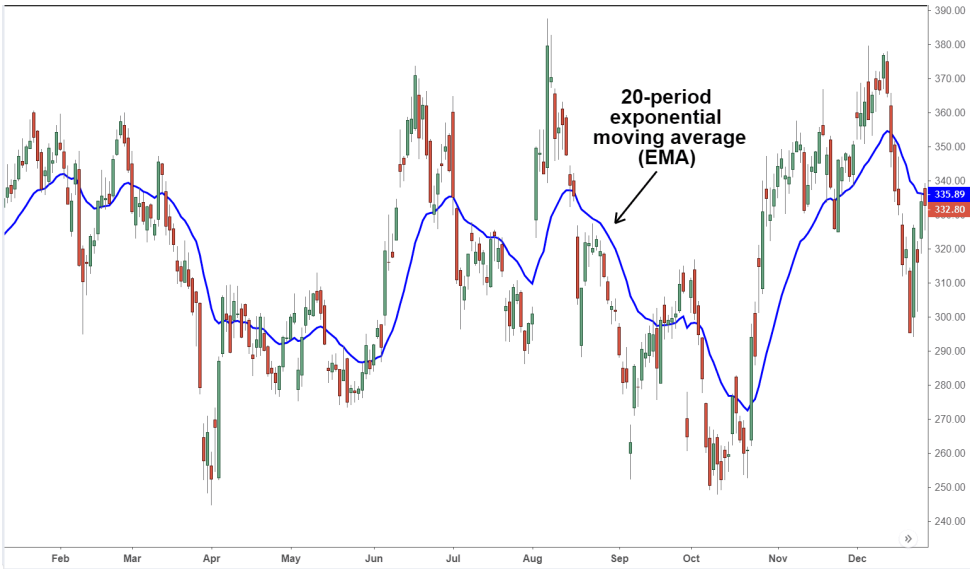


Fig. EMA Line corresponding to a particular stock.

When it comes to interpreting the EMA, when the market is in a continuous uptrend, the EMA indicator will also show an uptrend. The vice versa of this is true for a downtrend. The investor should not only pay attention to the direction of the line, but also the relative rate of change from one period to the next.

2. Volatility Ratio (VR)

Volatility ratio is an indicator that is used to identify price patterns and breakouts. This basically gives us the volatility of a stock. The volatility ratio is calculated using the following formula:

VR= ATR/TTR

where:

VR=Volatility Ratio

TTR (Today's True Range) = max (Today's High, Yesterday's Close) − min (Today's Low, Yesterday's Close)

ATR=Average True Range of the Past N-Day Period

The volatility ratio is a single line on the stock chart. A higher value of volatility ratio means that there is significant price volatility in the current trading day. This means that the price might face a change in trend either in the upwards or downwards direction. If the volatility ratio is less than 0.5, it means that the volatility is low. If the volatility ratio is greater than 0.5, it means that the stock price is headed for a breakout either in the positive or negative direction. Traders usually follow the VR indicator along with a few other trading indicators to help confirm a trading decision.

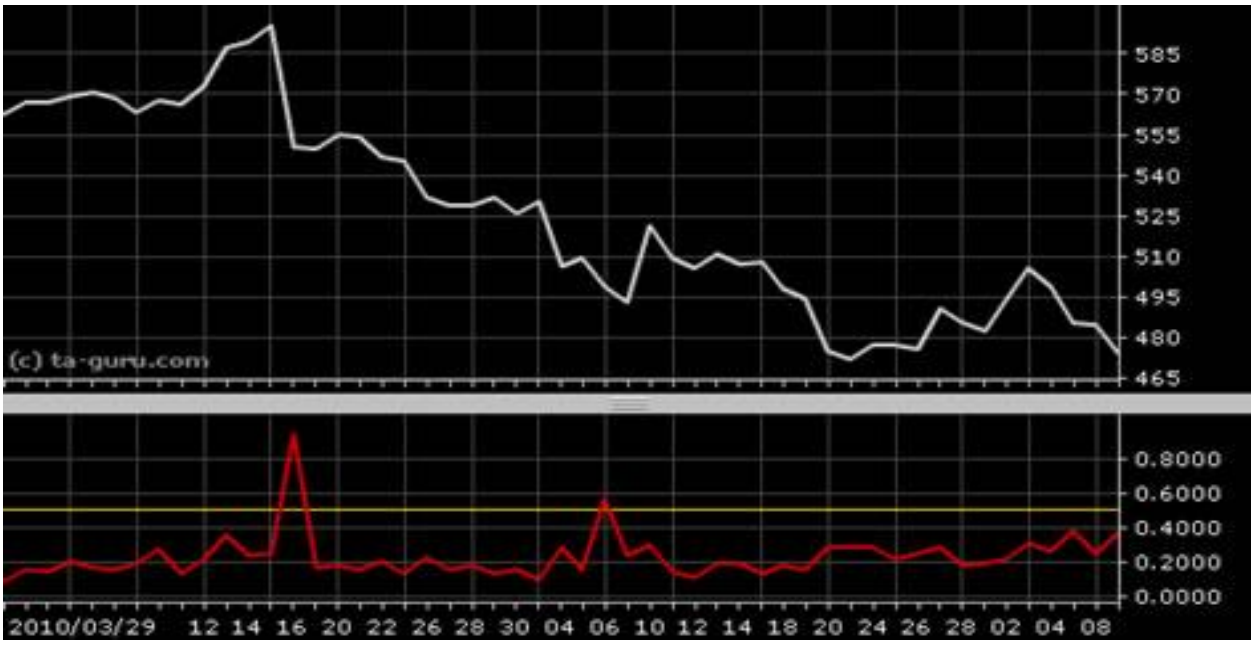


Fig. Variation in volatility ratio based on stock price.

3. William R %

Williams %R, also known as the Williams Percent Range, is a type of momentum indicator that moves between 0 and -100 and measures overbought and oversold levels. The Williams %R may be used to find entry and exit points in the market. It was developed by Larry Williams and it compares a stock's closing price to the high-low range over a specific period, typically 14 days or periods.

The Formula for the Williams %R is

Williams % R = Highest High – close / Highest High – Lowest Low

The indicator is telling a trader where the current price is relative to the highest high over the last 14 periods (or whatever number of lookback periods is chosen). When the indicator is between -20 and zero the price is overbought, or near the high of its recent price range. When the indicator is between -80 and -100 the price is oversold, or far from the high of its recent range. During an uptrend, traders can watch for the indicator to move below -80. When the price starts moving up, and the indicator moves back above -80, it could signal that the uptrend in price is starting again. The same concept could be used to find short trades in a downtrend. When the indicator is above -20, watch for the price to start falling along with the Williams %R moving back below -20 to signal a potential continuation of the downtrend. Traders can also watch for momentum failures. During a strong uptrend, the price will often reach -20 or above. If the indicator falls, and then can't get back above -20 before falling again, that signals that the upward price

momentum is in trouble and a bigger price decline could follow. The same concept applies to a downtrend. Readings of -80 or lower are often reached. When the indicator can no longer reach those low levels before moving higher it could indicate the price is going to head higher.

We apply the three indicators on the predicted value from the above sub section. We use a combination of all the three indicators to recommend the user to either buy, hold or sell a stock.

10. USER INTERFACE DESIGN AND IMPLEMENTATION

10.1 Start Page:

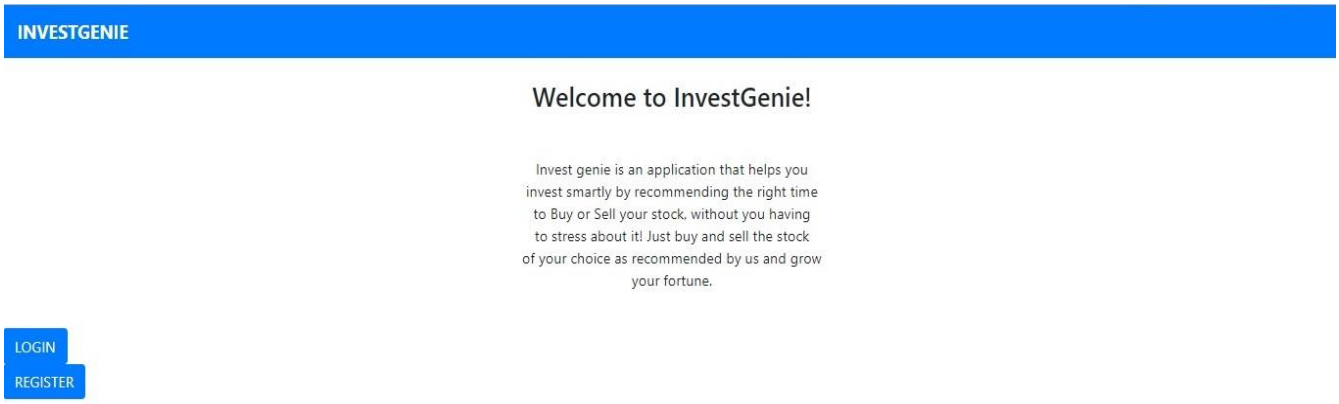


Fig. Start page of our web application.

This is the start page of our application. Here, a new user can register using the register button. If the user has already registered, he/she can go to Login page using the login button.

10.2 Register Page:

Fig. Register page of our web application.

This is the register page of our application. A new user can fill in the details, so that his/her credentials are saved in our database.

10.3 Login Page:

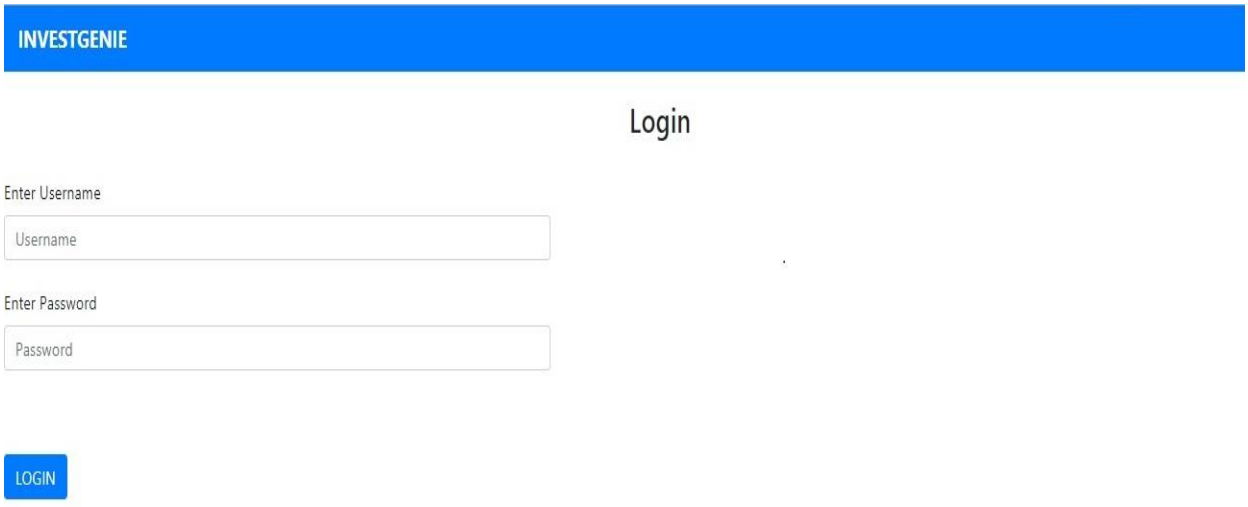


Fig. Login page of our web application.

This is the login page of our application. If the user has already registered, he/she can start using out web application by entering the required details and logging in.

10.4 Main Page:

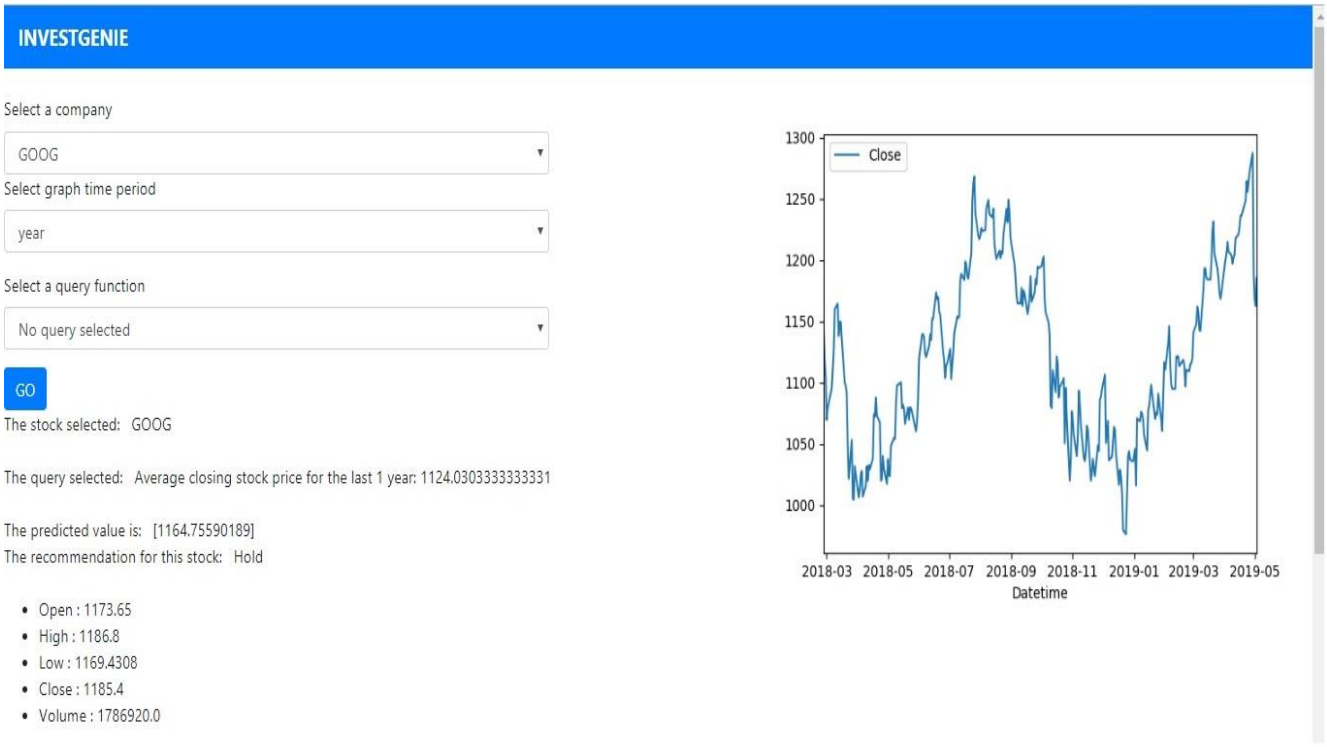


Fig. Main page of our web application.

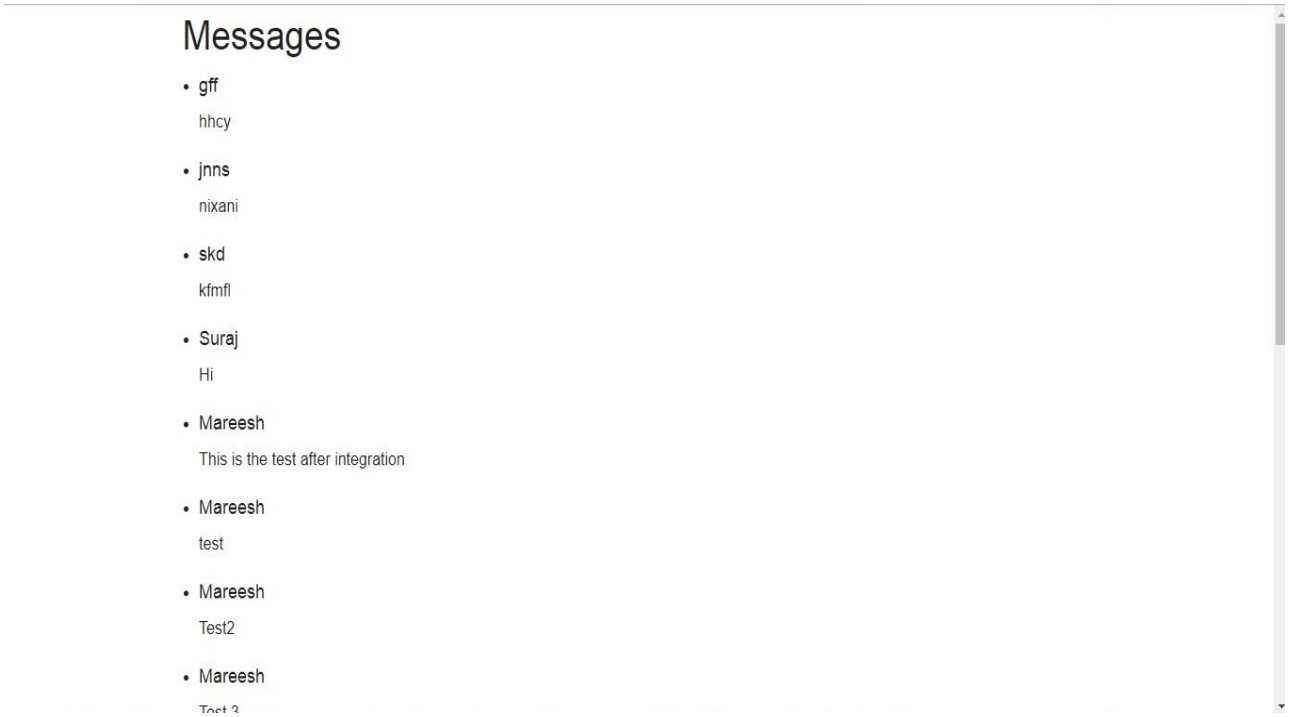
This is the main page of our application. Here, using a drop down list the user can select a stock symbol (from a predefined list of 10 symbols), select the duration for which he needs the information, and select the type of query (from a predefined ser of 5 queries). After selecting the desired options and pressing the enter key, the user can see the output of his query on the same page as shown in the figure above.

Fig. Main page of our web application.

The main page of our application has a section for comments at the end of the page.

User can enter their name and their comments here.

10.5 Comments:



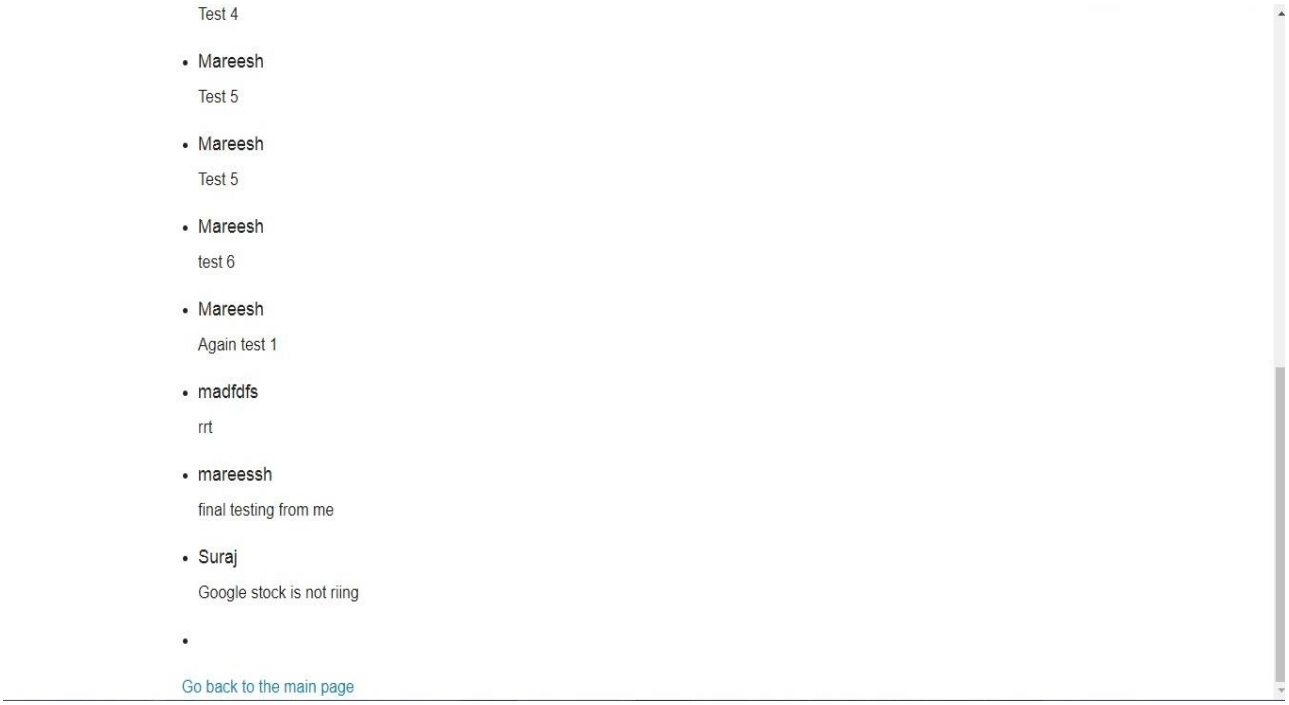


Fig. Comments page of our web application.

This is the comments page of our application. After a user enters some comments and presses the submit button, they will be directed to this page. This page shows all the comments in chronological order and has an option at the end to return to the main page.

11.FUTURE WORK

As a part of the future work for this project, the user experience of the application can be improved in terms of display of dynamic graphs with which the users can interact. Also, the technical indicators have to be displayed on the graph along with the historic stock data. The website can also be made simpler to use in terms of comments being posted in the same page and scaling of the various items on the page has to be made more user friendly.

Secondly, the number of features that the website boasts of can be increased. Features such as subscribing to stocks, getting alerts if it is the right time to buy/sell a

stock that a user has subscribed to, etc. Many such user specific features can be added so that each user of the application has a personalized experience.

Another thing to be optimized is the entire system architecture to ensure that the wait times for users are minimal and the overall load on the server CPU and memory are reduced.

12.ACKNOWLEDGEMENT

We would like to thank our Professor Shiyu Zhou and the graders Weijia Sun and Huayu Zhao for their assistance and guidance throughout the entire course.

13.REFERENCES

[1] https://towardsdatascience.com/creating-a-dataset-using-an-api-with-python-dcc1607616d

[2] https://en.wikipedia.org/wiki/Bootstrap_(front-end_framework)

[3] https://getbootstrap.com/

[4] https://www.mysql.com/products/workbench/

[5] https://stackify.com/soap-vs-rest/

[6] https://blog.miguelgrinberg.com/post/designing-a-restful-api-with-python-and-flask

[7] https://en.wikipedia.org/wiki/Flask_(web_framework)

[8] https://www.investopedia.com/ask/answers/122314/what-exponential-moving-average-ema-formula-and-how-ema-calculated.asp

[9] https://www.investopedia.com/terms/v/volatility-ratio.asp

[10] https://www.investopedia.com/terms/w/williamsr.asp

[11] https://towardsdatascience.com/introduction-to-bayesian-linear-regression-e66e60791ea7

[12] https://scikit-learn.org/stable/modules/linear_model.html#bayesian-ridge-regression

[13] https://scikit-learn.org/stable/modules/linear_model.html#bayesian-ridge-regression

[14] https://en.wikipedia.org/wiki/Support-vector_machine

[15] https://www.saedsayad.com/support_vector_machine_reg.htm

[16] https://skymind.ai/wiki/neural-network

[17] https://skymind.ai/wiki/multilayer-perceptron