RU Rich Stock Forecaster system

(Web-Based Stock Forecaster)

SOFTWARE ENGINNERING OF WEB APPLICATIONS

Team: Group # 7

Team Member:

Zhongze Tang (zt67),

Zichen Zhu (zz313),

Weijia Sun (ws368),

Weitian Li (wl436),

Huayu Zhao (hz274)

2018.4.26

1 Contribution Breakdown

We all contribute equally in this project, and our project goes well under the strength of our efforts. Each one contributes equally and we have finished a great project. The work everyone did are listed as below:

Front-end: Huayu Zhao

Back-end: Weijia Sun

Prediction Engine & SA: Zichen Zhu & Zhongze Tang

Testing & Document: Weitian Li

1.1 Costumer Statement of Requirements

Stock investment refers to the behavior of a company or individual to purchase stocks in an accumulated currency to obtain income. The income from stock investment is composed of two parts that are revenue income and capital gain. Revenue income refers to the dividends and dividends received by stock investors in the company's profit distribution according to their shareholdings. Capital gains refer to the gains that investors receive in the course of changes in stock prices that is, buying stocks at low prices and selling them at high prices. When the stock market first appeared in the world, it becomes the main way for businessmen to make capital profit.

1.2 Motivation

Nowadays, people are more and more likely to seek an effective and reliable way to maintain their personal financial investment in wealth growth. Investment in the stock market becomes a popular way for public to manage their wealth, since making high profit in the stock market seen easier and the popularity of the Internet and Personal Computers. Everyone can participate in the stock market. The entry threshold of stock investing is low, and most of investor pursuing making higher and higher profit. However, high profit brings high risks. People need to make prudent and wise decisions to reduce risks. It's clear that most of the unprofessional investor lose their money in the stock market. The reason is as follow. First, most of unprofessional investor don't spend lot of time caring the stock price and related value. Second, most of unprofessional investor don't have enough stock and financial investment knowledge. With the development with Internet and PC, currently, people are able to make their deal and manage stock sharer and fund on the computer, which need specialized agencies in the past. For most unprofessional investors, their purpose is to make profit and try their best to avoid the unpresented risk. But they lack a general and smart tool to help them to make decision compare with the professional investors like big companies and stock exchange agencies. Also, they don't have enough time and knowledge to help them to distinguish which stock will have potential to have higher price or lower price. Most of them are in

professional jobs in the cities.

1.3 Vision

Thanks to advanced Internet and programming tool. Computer can provide timely stock information and related auxiliary information for people to make wise decision in the investments. The technical analysis is based on the past and present market behavior of the securities market as the analysis object, using mathematical and logical methods to explore some typical rules of change in order to predict the future trend of the market. Through the use of various securities analysis tools and indicators to simulate the trend of the real stock market, more investors can understand the basic methods of the stock market and make better decision on their stock. So, we develop a web-based stock system for users to find the price of given stock and make prediction. In our system, there are three levels. The first is user interface level for showing the price and prediction and input stream. The second is back-end level where storing data and processing data. The third one is algorithms engine, which is for predicting stock price and making prediction strategies. User can find the given stock price on our website. When the user chooses wanted stock and wants to seek for more information, our website will send a request to back-end level looking for data and prediction. The back-end level will send a request to algorithms engine asking for predicted price and strategy of given stock, and after that,

back-end level will send above information and historical stock price data to the user interface level. Receiving the data, user interface can plot the stock price graphs and show the strategies and indicators. In our website, users can find the historical and predicted price of given stock. Also, users can find the strategies and comments from past users about given stock. Option: Stock List, Historical Stock Data, Comment, Indicators and Price Prediction.

1.1.1. Stock List

It will select 10 stocks and show their names and real time price on the left table. We select these from https://www.alphavantage.co/.

Historical Stock Data

It will provide the graph of month and year price data of stock in the stock list for users to query.

Comment

It will provide users to input their comment about wanted stock in the stock list.

Indicators

It will provide three indicators for users to judge the trend of the stock. We provide EMA, VR and EMCD.

Price Prediction

It will use three methods to predict the price of stock and average three predicted price and get the final predict price. We use Bayesian Curve Fitting, ANN and SVM.

Share

It will provide users to share their opinion and the link of page of wanted stock in Twitter, G+ and LinkedIn.

2 GLOSSARY OF TERMS

Algorithms: a step-by step procedure for calculations.

Artificial Neural Network: Neural networks is the development of a computer system that models the human brain and its nervous system. also see (Artificial intelligence)

Buy And Hold: The acquisition of a tradable good for the long term rather than trying to profit over a quick turnover.

Charts: A image or display of a stock/security that plots price and/or volume (the number of shares sold).

Closing price: the final price at which a security is traded on a given trading day. It represents the most up-to-date valuation of a security until the next

trading day.

Database: an organized collection of data that are typically organized to model relevant aspects of reality in a way that supports process rewiring this information.

Machine learning: A branch of artificial intelligence, concerned with the construction and study of systems that can learn from data.

Neural network: conceptually based off the central nervous system, it interconnects systems of neurons that can calculate values for inputs by feeding information through the network.

Fundamental analysis: The method of evaluating a security in an attempt to measure its intrinsic value, by examining related economic, financial and other qualitative and quantitative factors. Fundamental analysts study anything that can affect the security's value, including macroeconomic factors such as the overall economy and industry conditions, and microeconomic factors such as financial conditions and company management.

Web service: is a service offered by an electronic device to another electronic device, communicating with each other via the World Wide Web. In a web service, the Web technology such as HTTP—originally designed for human-to-machine communication—is utilized for machine-to-machine communication, more specifically for transferring machine-readable file formats such as XML and JSON.4.

3 SYSTEM REQUIREMENTS

3.1 ENUMERATED FUNCTIONAL REQUIREMENTS

Identifier	Priority	Requirement
REQ1	5	The system will allow system administrators to add or remove stocks from the stock lists.
REQ2	5	The system will constantly collect historical and real time data of the stock in the stock lists.
REQ3	4	The system will print the stock historical and real time prices as a graph along with the timeline.
REQ4	4	The system will allow user to choose stock in given stock list.
REQ5	4	The system will allow user to input their comment on wanted stock in given stock list and see the comment.
REQ6	4	The system will periodically apply prediction algorithms or models on the obtained data and store the results to a central database.
REQ7	3	The system should store a list of stocks that the user chooses in the database.

REQ8	3	The system should pull the data from the back-end and show the stock as the users wanting.
REQ9	3	The system should provide indicators' data and prediction strategy.
REQ10	2	The system should provide unique page for each stock in the given stock list.

3.2 ON-SCREEN APPEARANCE REQUIREMENTS

Identifier	Priority	Requirement
REQ11	5	A fixed navigation bar that includes all of the functional features, allowing the user to quickly navigate to any feature that they may choose without going through various subpages and menus.
REQ12	5	The page must adapt to a various size in order to maintain functionality, consistent design, as well as ease of use on a mobile device
REQ13	4	The website must support the latest versions of the most popular browsers, Google Chrome, Firefox, and Internet Explorer.
REQ14	4	Function and purpose of each element on screen must be clear and direct by placing

		information in natural areas where natural will be defined based on other popular stock websites, as well as developer/designer intuition

4 FUNCTIONAL REQUIREMENTS SPECIFICATION

4.1 Stakeholders

Two Stakeholders can be identified:

1. User: any user could log in the system and get web services.

2. Administrator: maintains and updates website services.

4.2 Actor and Goals

In our system, it has two kinds of actor.

User: an ordinary user.

Administrator: The manager that is in charge or keeping the system updated and in working order.

Prediction Algorithm: The algorithm(s) that will calculate the prediction.

Database: The database will hold all the user data and information as well

as all the stock information. All the user information and their portfolios will be stored in the database.

Alpha Vantage API: The API is where we will pool the real time stock data.

Graph: Provide visual charts from raw data.

Alpha Vantage: This is where we will pool out the historical stock data.

4.3 Use Cases

Use Case	**User priority**	**Function**
Case 1	Administrator	Manage RU RICH System
Case 2	User	Stock and price Query
Case 3	User	Plot special graph
Case 4	User	Stock Indicator
Case 5	User	Price Prediction
Case 6	User	Comment
Case 7	User	Share
Case 8	User	Get Quotes

Case 1: Manage RU RICH System

As the administrators, they need to analysis whether the website is work normally. And they also need to manage and maintain the web serving the

customers continually. As an important part of the whole system, the administrator can create the database, update the data in the database (send a request to Google or Yahoo for the data and read them to import them to the database), and manage the users' information.

Case 2: Stock and price Query

For each main option (interested, hold), there are several stocks stored inside.

The customers can get a lot of information of these stocks, such as the real-time price, the historical price, the recent price graphs (days, months, one years, two years) and so on. This provides user with information about the company and the last year's all quarter results and the current years' quarter results.

Case 3: Plot special graph

As a presentation of history price and current price, the user can notice the flow and trend of the stocks in the graph. The user can click "Price Data" then "historical price" or "real time" to access the result page, then choose a stock to check the specific graph of the stock. The page will call a function to get the data from the database then show them on the screen as a chart.

Case 4: Stock Indicator

After analyzing the history data and prediction, the server should return back a guiding suggestion to the user. Based on the prediction result, the

server can analyze the user's performance and market situation to get some guiding result and help the user to do the decision.

Case 5: Price Prediction

For the prediction of each stock, users do not care how you predict the prices, the only thing they care about is the accuracy of your prediction. Hence, it is necessary to provide the data accuracy for each stock's prediction to enable users know the whether they should trust a specific predication.

Case 6: Comment

For the comment function, users can see the price and indictors of wanted stock and have some words to say. They can leave their opinions in the following input box. Their words and time will be recorded and show in the bottom of the stock information.

Case 7: Share

For the share function, users can share opinions with orders through Twitter, G+ and LinkedIn.

Case 8: Get Quotes

After logging in the system, the user can get the quotes of the stocks. At the index page of the system, the user can click the "stock list" to access the stock list page. There will be the stock symbols with their real-time price.

In this case, the website will send a request to the database and catch the current price and show them on the screen.

5 EFFORT ESTIMATION

5.1 Unadjusted Actor Weight

Actor Name	Description	Complexity	Weight
Administrator	Special case User that maintains and updates website services.	Complex	3
Prediction Algorithm	Provide prediction indicators and strategy.	Complex	3
Database	Records of stock information, user data, and system data	Complex	3
User	To use our system.	Average	2
API	Provide real time stock price and historical stock price.	Complex	3
Graph	Provide visual charts from raw data.	Average	2

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

5.2 Unadjusted Use Case

Use Case	Complexity	Function	Weight
Case 1	Complex	Manage RU RICH System	9
Case 2	Complex	Stock and price Query	9
Case 3	Complex	Plot special graph	9
Case 4	Complex	Stock Indicator	9
Case 5	Complex	Price Prediction	9
Case 6	Average	Comment	6
Case 7	Average	Share	6
Case 8	Simple	Get Quotes	3

UUCW=9+9+9+9+9+6+6+3=60

UUCP=UAW+UUCW=16+60=76

6 Domain Analysis

6.1 Concept Definitions

The concept and their definitions are discussed below.

Website:

Definition: A hypertext document connected to the World Wide Web.

Responsibilities:

 Display HTML document that shows the actor the current context.

 Shows what actions can be taken through buttons.

Query：

Definition: Search query.

Responsibilities:

 Hold a specific search query.

PageMaker:

Definition: Generates display inputs ultimately for website

Responsibilities:

 Must be able to display text, numbers and graphics for website environment

Motivation: Data and images simply cannot come to website in a quick and easy fashion. There must be a transformation or parsing of “raw” local data that can be manipulated to fit the website environment.

Predictor:

Definition: Generate stock predictions.

Responsibilities:

 Apply prediction algorithm to data.

Update Historical:

Definition: Send a request to the data provider and fetch the historical price data.

Responsibilities:

 Retains momentary stock data from external websites and passes to Data Handler.

Timer:

Definition: A function to calculate internal time internal and refresh.

Responsibilities:

Store time and refresh the page.

Database Connection:

Definition: An organized collection of stock data and system data.

Responsibilities:

 Store times.

 Store stock data.

6.2 Attributes

Website holds attributes related to display text, graph and related API to social media.

Query holds attributes related to search query for sector, industry and

keyword. In our system, we can use keyword.

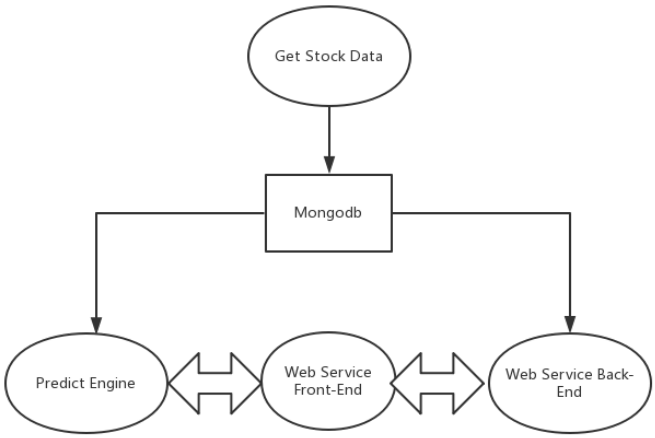
Database: Connection holds attributes to connect to the local database including address, port number, password and user name.

Timer holds attributes related to when to update current stock prices as well as the time when to predict future stock prices.

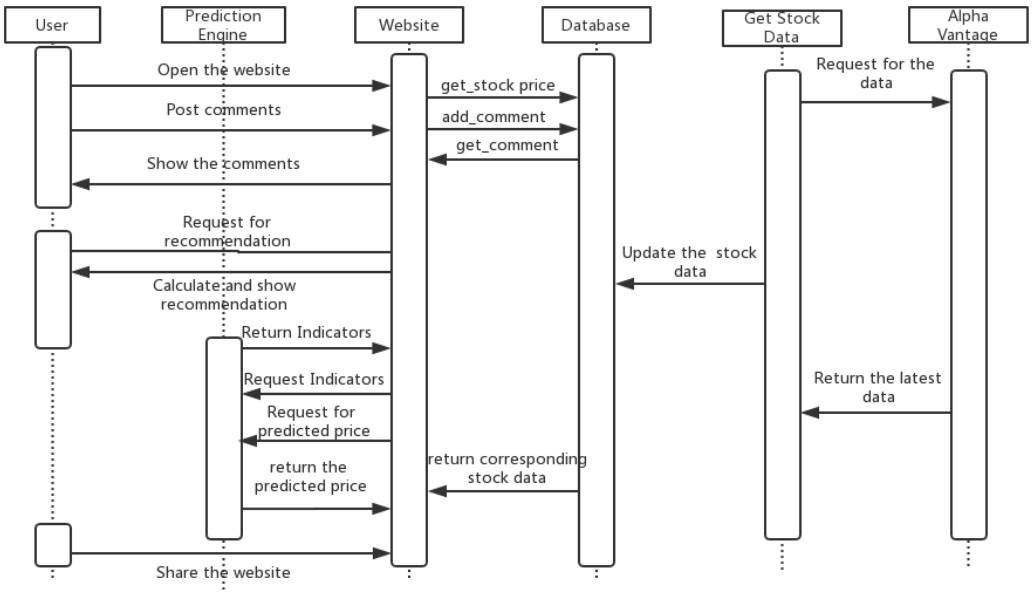
7 Class Diagram and Interface Specification

7.1 Class Diagram

Overview



Time sequence diagrams:



7.2 Data Types and Operation Signatures

Predictor

- prediction_engine.bayes.Bayes.predict(X: np.ndarray, y: np.ndarray, x: np.ndarray): array

help user predict the price of the stock they want with Bayesian Linear Regression. The first element of the array is the prediction price.

- predicton_engine.svr_zhu.SupportVectorRegression.predict(X: np.ndarray, y: np.ndarray, x: np.ndarray): array

help user predict the price of the stock they want with Support Vector Regression. The first element of the array is the prediction price.

- prediction_engine.dnn.DNN.predict(X: np.ndarray, y: np.ndarray, x:

np.ndarray): double

help user predict the price of the stock they want with Deep Neural Network.

- get_advice(var VRlast, var EMAlast, var MACDlast): void

After providing the indicators of each stock, we will also show our suggestion that the user should hold, sell or buy the stock.

- getAllStock(): List<RealtimeStock>

Get the list of all companies in the database along with their latest stock price (real time latest stock price)

- getMax(String symbol): JSON

Get the highest stock price of any company in the last ten days

- getMin(String symbol): JSON

Get lowest stock price for any company in the latest one year.

- getAvg(String symbol): JSON

Get average stock price for any company in the latest one year.

- getComment(String symbol): JSON

Get the comment, username and the timestamp of the corresponding stock.

- addComment(String symbol, String comment, String timestamp, String username): JSON

Add the comment with the username, timestamp and the corresponding symbol name into the database.

- getDay(String symbol): JSON

Get the data for plotting the figure within one day range.

- getMonth(String symbol): JSON

Get the data for plotting the figure within one month range.

- getYear(String symbol): JSON

Get the data for plotting the figure within one year range.

Indicator

- prediction_engine.vr.VolatilityRatio.indicator(price: np.float_, historical_price: np.ndarray, historical_volume: np.ndarray): np.float_

calculate the value of VR indicator.

- prediction_engine.ema.EMA.value(vals: np.ndarray): np.float_

calculate the value of EMA indicator.

- prediction_engine.macd.MACD.value(val12: np.ndarray, val26:np.ndarray): np.float_

calculate the value of MACD indicator.

8 System Architecture and System Design

8.1 Architecture Styles

Bootstrap

Bootstrap is a free and open-source front-end web framework for designing websites and web applications. It contains HTML- and CSS-based design templates for typography, forms, buttons, navigation and other interface

components, as well as optional JavaScript extensions. Unlike many web frameworks, it concerns itself with front-end development only. In the design of the user interface, Bootstrap is widely used to keep the web page dynamically showing suitable to the screen of the user.

jQuery

jQuery is a cross-platform JavaScript library designed to simplify the client-side scripting of HTML. It is free, open-source software using the permissive MIT license. Web analysis indicates that it is the most widely deployed JavaScript library by a large margin. jQuery's syntax is designed to make it easier to navigate a document, select DOM elements, create animations, handle events, and develop Ajax applications. jQuery also provides capabilities for developers to create plug-ins on top of the JavaScript library. This enables developers to create abstractions for low-level interaction and animation, advanced effects and high-level, they eable widgets. The modular approach to the jQuery library allows the creation of powerful dynamic web pages and Web applications. jQuery is used to let the client easier to pass the data and create the chart.

8.2 Data Storage

In data storage part, we use MongoDB to create database and store the data. There are three tables in our database, they are Historical price and

Realtime Prices.

HistoricalPrice

Historical is used to store the historical price data of the stocks and their indicators' parameter. Here is a figure to show the structure of the table.

Collection: daily

Name	Type	Description
timestamp	date	The date and time of record. Stored in UTC timezone
symbol	string	Name of stock
open	decimal128	Open price
high	decimal128	High price
low	decimal128	Low price
close	decimal128	Close price
volume	int64	Volume of stock

RealtimePrices

Realtime prices is a table which is used to store the real-time price and volume data of the stocks. It will be updated automatically every minute and insert a new value into the table.

Here is the structure of the table.

Collection: realtime

Name	Type	Description
timestamp	date	The date and time of record. Stored in UTC timezone
symbol	string	Name of stock
price	decimal128	The real time price of the stock
volume	int64	Volume of stock

Comment

Collection: comment

Name	Type	Description
timestamp	date	The date and time of record. Stored in UTC timezone
symbol	string	Name of stock
username	string	Username of the user
comment	string	The content of the comment

8.3 Web Service

RESTFUL

REpresentational State Transfer (REST) is an architectural style that defines a set of constraints and properties based on HTTP. Web Services that conform to the REST architectural style, or RESTful web services, provide interoperability between computer systems on the Internet. REST-compliant web services allow the requesting systems to access and manipulate textual representations of web resources by using a uniform and predefined set of stateless operations. Other kinds of web services, such as SOAP web services, expose their own arbitrary sets of operations.

"Web resources" were first defined on the World Wide Web as documents or files identified by their URLs. However, today they have a much more generic and abstract definition that encompasses every thing or entity that can be identified, named, addressed, or handled, in any way whatsoever, on the web. In a RESTful web service, requests made to a resource's URI will elicit a response that may be in XML, HTML, JSON, or some other format. The response may confirm that some alteration has been made to the stored resource, and the response may provide hypertext links to other related resources or collections of resources. When HTTP is used, as is most common, the operations available are GET, POST, PUT, DELETE, and other predefined CRUD HTTP methods.

By using a stateless protocol and standard operations, REST systems aim

for fast performance, reliability, and the ability to grow, by re-using components that can be managed and updated without affecting the system as a whole, even while it is running.

RESTful API Design Definitions

Resource: A single instance of an object. For example, an animal.

Collection: A collection of homogeneous objects. For example, animals.

HTTP: A protocol for communicating over a network.

Consumer: A client computer application capable of making HTTP requests.

Third Party Developer: A developer not a part of your project but who wishes to consume your data.

Server: An HTTP server/application accessible from a Consumer over a network.

Endpoint: An API URL on a Server which represents either a Resource or an entire Collection.

Idempotent: Side-effect free, can happen multiple times without penalty.

URL Segment: A slash-separated piece of information in the URL.

Flask

Flask is a micro web framework written in Python and based on the Werkzeug toolkit and Jinja2 template engine. It is BSD licensed.

The latest stable version of Flask is 0.12.2 as of May 2017. Applications that use the Flask framework include Pinterest, LinkedIn, and the

community web page for Flask itself.

Flask is called a micro framework because it does not require particular tools or libraries. It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions. However, Flask supports extensions that can add application features as if they were implemented in Flask itself. Extensions exist for object-relational mappers, form validation, upload handling, various open authentication technologies and several common framework related tools. Extensions are updated far more regularly than the core Flask program.

8.4 Algorithms

Scikit-learn

Scikit-learn provides a range of supervised and unsupervised learning algorithms via a consistent interface in Python.

It is licensed under a permissive simplified BSD license and is distributed under many Linux distributions, encouraging academic and commercial use.

The library is built upon the SciPy (Scientific Python) that must be installed before you can use scikit-learn. This stack that includes:

NumPy: Base n-dimensional array package

SciPy: Fundamental library for scientific computing

Matplotlib: Comprehensive 2D/3D plotting

IPython: Enhanced interactive console

Sympy: Symbolic mathematics

Pandas: Data structures and analysis

Extensions or modules for SciPy care conventionally named SciKits. As such, the module provides learning algorithms and is named scikit-learn. The vision for the library is a level of robustness and support required for use in production systems. This means a deep focus on concerns such as easy of use, code quality, collaboration, documentation and performance. Although the interface is Python, c-libraries are leverage for performance such as numpy for arrays and matrix operations, LAPACK, LibSVM and the careful use of cython.

Tensorflow

TensorFlow™ is an open source software library for high performance numerical computation. Its flexible architecture allows easy deployment of computation across a variety of platforms (CPUs, GPUs, TPUs), and from desktops to clusters of servers to mobile and edge devices. Originally developed by researchers and engineers from the Google Brain team within Google's AI organization, it comes with strong support for machine learning and deep learning and the flexible numerical computation core is used across many other scientific domains.

9 ALGORITHMS AND DATA STRUCTURES

9.1 Bayesian Theory

In probability theory and statistics, Bayes' theorem (alternatively Bayes' law or Bayes' rule, also written as Bayes's theorem) describes the probability of an event, based on prior knowledge of conditions that might be related to the event.

Bayes' theorem is stated mathematically as the following equation:

$$P(A \mid B) = \frac{P(B \mid A)\,P(A)}{P(B)}$$

where A and B are events and $P(B) \neq 0$.

P(A) and P(B) are the probabilities of A and B independent of each other. P(A | B), the conditional probability of A given that B is true. P(B | A) is the probability of B given that A is true.

9.2 Curve fitting

Curve fitting is the process of constructing a curve, or mathematical function, that has the best fit to a series of data points, possibly subject to constraints. Curve fitting can involve either interpolation, where an exact fit to the data is required, or smoothing, in which a "smooth" function is

constructed that approximately fits the data. A related topic is regression analysis, which focuses more on questions of statistical inference such as how much uncertainty is present in a curve that is fit to data observed with random errors. Fitted curves can be used as an aid for data visualization, to infer values of a function where no data are available, and to summarize the relationships among two or more variables. Extrapolation refers to the use of a fitted curve beyond the range of the observed data, and is subject to a degree of uncertainty since it may reflect the method used to construct the curve as much as it reflects the observed data.

9.3 Bayesian Curve Fitting

Concept and Definition

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:

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

- 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 $\phi(x)=(1,x,x2,...,xM)T$.

9.4 support vector machine (SVM)

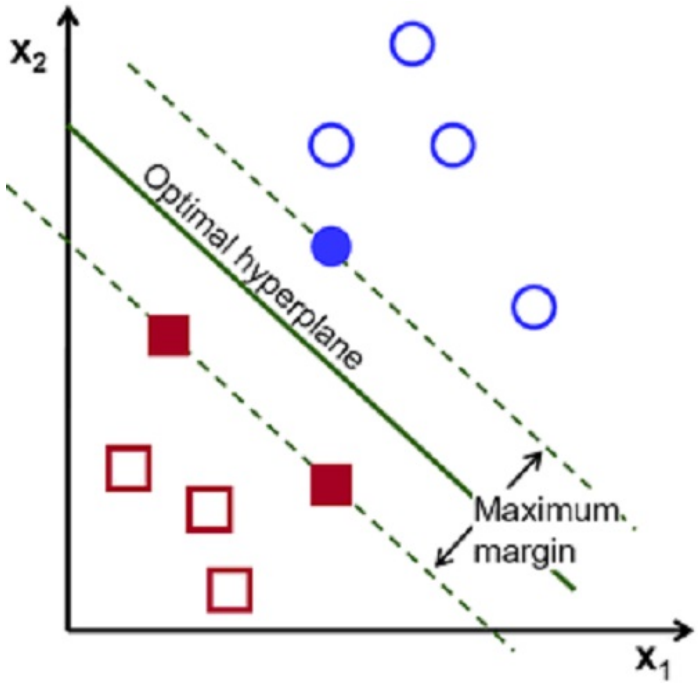
Concept and Definition

support vector machines (SVMs, also support vector networks) are supervised learning models with associated learning algorithms that analyze data used for classification and regression analysis. Given a set of training examples, each marked as belonging to one or the other of two categories, an SVM training algorithm builds a model that assigns new

examples to one category or the other, making it a non-probabilistic binary linear classifier (although methods such as Platt scaling exist to use SVM in a probabilistic classification setting). An SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall.

In addition to performing linear classification, SVMs can efficiently perform a non-linear classification using what is called the kernel trick, implicitly mapping their inputs into high-dimensional feature spaces.

When data are not labeled, supervised learning is not possible, and an unsupervised learning approach is required, which attempts to find natural clustering of the data to groups, and then map new data to these formed groups. The support vector clustering algorithm created by Hava Siegelmann and Vladimir Vapnik, applies the statistics of support vectors, developed in the support vector machines algorithm, to categorize unlabeled data, and is one of the most widely used clustering algorithms in industrial applications.

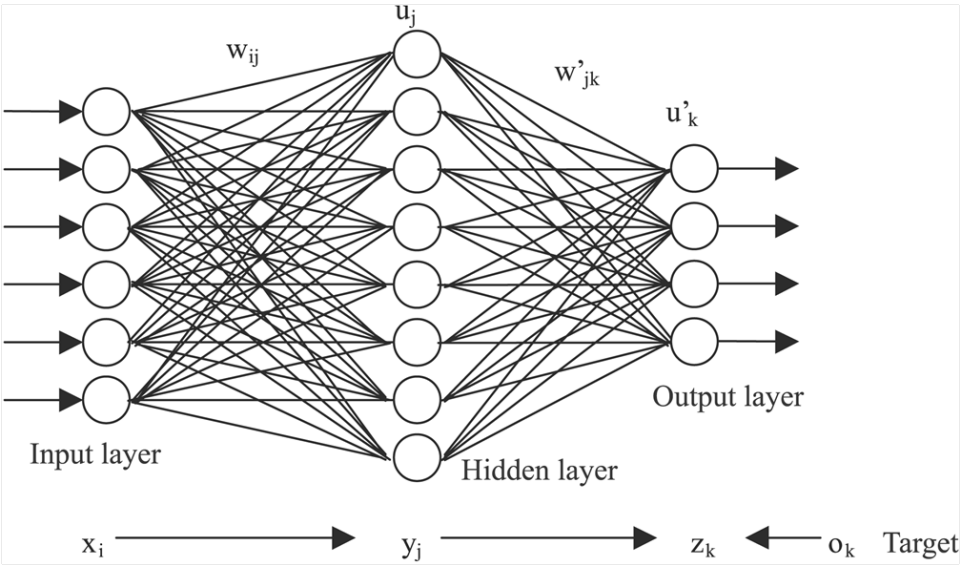


9.5 Artificial neural network

Concept and Definition

The word network in the term ‘artificial neural network’ refers to the interconnections between the neurons in the different layers of each system. An example system has three layers. The first layer has input neurons which send data via synapses to the second layer of neurons, and then via more synapses to the third layer of output neurons. More complex systems will have more layers of neurons, some having increased layers of input neurons and output neurons. The synapses store parameters called "weights" that manipulate the data in the calculations. Artificial neural networks (ANNs) or connectionist systems are computing systems vaguely inspired by the biological neural networks that constitute animal brains. Such systems

"learn" (i.e. progressively improve performance on) tasks by considering examples, generally without task-specific programming. For example, in image recognition, they might learn to identify images that contain cats by analyzing example images that have been manually labeled as "cat" or "no cat" and using the results to identify cats in other images. They do this without any a priori knowledgeabout cats, e.g., that they have fur, tails, whiskers and cat-like faces. Instead, they evolve their own set of relevant characteristics from the learning material that they process.



Programming Technologies

We totally have 4 layers which means we set two hidden layer together with input layer and output layer.

Short term: 50 days' prices as 50 inputs, 20 hidden neurons for each hidden layer and one output neuron.

Long term: 200 days' prices as 200 inputs, 20 hidden neurons for each hidden layer and one output neuron.

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.

Training Set: 50 days' prices before wanted days in short term;

200 days' prices before wanted days in long term.

After trained, we will choose 50 days' price to predict price in short term and 200 days' price to predict price in long term.

9.6 MACD Indicator

Definition:

Developed by Gerald Appel in the late seventies, the Moving Average Convergence/Divergence oscillator (MACD) is one of the simplest and most effective momentum indicators available. The MACD turns two trend-following indicators, moving averages, into a momentum oscillator by subtracting the longer moving average from the shorter moving average. As a result, the MACD offers the best of both worlds: trend following and

momentum. The MACD fluctuates above and below the zero line as the moving averages converge, cross and diverge. Traders can look for signal line crossovers, centerline crossovers and divergences to generate signals. Because the MACD is unbounded, it is not particularly useful for identifying overbought and oversold levels.

Calculation:

MACD Line: (12-day EMA - 26-day EMA)

Signal Line: 9-day EMA of MACD Line

MACD Histogram: MACD Line - Signal Line

the MACD is all about the convergence and divergence of the two moving averages. Convergence occurs when the moving averages move towards each other. Divergence occurs when the moving averages move away from each other. The shorter moving average (12-day) is faster and responsible for most MACD movements. The longer moving average (26-day) is slower and less reactive to price changes in the underlying security.

Working Process:

The MACD Line oscillates above and below the zero line, which is also known as the centerline. These crossovers signal that the 12-day EMA has crossed the 26-day EMA. The direction, of course, depends on the direction of the moving average cross. Positive MACD indicates that the 12-day EMA is above the 26-day EMA. Positive values increase as the shorter EMA diverges further from the longer EMA. This means upside momentum

is increasing. Negative MACD values indicate that the 12-day EMA is below the 26-day EMA. Negative values increase as the shorter EMA diverges further below the longer EMA. This means downside momentum is increasing.

Programming Technologies:

We use Python and MongoDB to calculate the MACD indicator. First, we use Python to query the latest 12 and 26 prices of stocks from MongoDB and send it to our program. Then our program will call the EMA function to calculate the 12-day EMA and 26-day EMA. Finally, we use the formula 12-day EMA – 26-day EMA to get the value of MACD.

9.7 VR Indicator

Definition:

The Volatility Ratio is a technical analysis indicator used to detect wide-ranging days, days with an unusual distance between the high and low prices.

The Volatility Ratio is based on the True Range (TR) indicator and it is computed by dividing the current true range value by the N-Bar exponential moving average of the true range. The true range is the highest value among: today's high minus low, today's high minus yesterday's close and yesterday's close minus today's low.

Calculation:

Current True Range = Maximum (average of current day's high and yesterday's close) - Minimum (average of today's low and yesterday's close)

Previous True Range over X number of days = HIGH (average of the high prices of each day over time period X) - LOW (average of the low prices of each day over time period X)
Volatility Ratio = Current True Range / Previous True Range over X number of days

Working Process:

Usually, traders consider that ranging day signals happen when volatility ratio is higher than 2. These signals may occur after price gaps or wide-ranging days and could indicate a likely reversal. The volatility ratio is generally used in combination with other trading indicators, such as a volume indicator, to confirm or not the breakout/reversal.
The Volatility Ratio indicator name is "volratio" and it contains one parameter which is the period used to calculate the exponential moving average of the true range. The default volatility ratio period is set to 14.

Programming Technologies:

We use Python and MongoDB to calculate the VR indicator.

9.8 EMA Indicator

Definition:

Exponential moving average (EMA) is a type of infinite impulse

responsefilter that applies weighting factors which decrease exponentially. The weighting for each older datum decreases exponentially, never reaching zero. The graph at right shows an example of the weight decrease.

Calculation:

The EMA for a series Y may be calculated recursively

$$S_t = \begin{cases} Y_1, & t = 1 \\ \alpha \cdot Y_t + (1-\alpha) \cdot S_{t-1}, & t > 1 \end{cases}$$

Where:

- The coefficient α represents the degree of weighting decrease, a constant smoothing factor between 0 and 1. A higher α discounts older observations faster.
- Y_t is the value at a time period t.
- S_t is the value of the EMA at any time period t.

Sample:

Moving averages smooth the price data to form a trend following indicator. They do not predict price direction, but rather define the current direction

with a lag. Moving averages lag because they are based on past prices. Despite this lag, moving averages help smooth price action and filter out the noise. They also form the building blocks for many other technical indicators and overlays, such as Bollinger Bands, MACD and the McClellan Oscillator. The two most popular types of moving averages are the Simple Moving Average (SMA) and the Exponential Moving Average (EMA). These moving averages can be used to identify the direction of the trend or define potential support and resistance levels.

Programming Technologies:

We use Python and MongoDB to calculate the EMA indicator.

9.9 Final Prediction Strategy

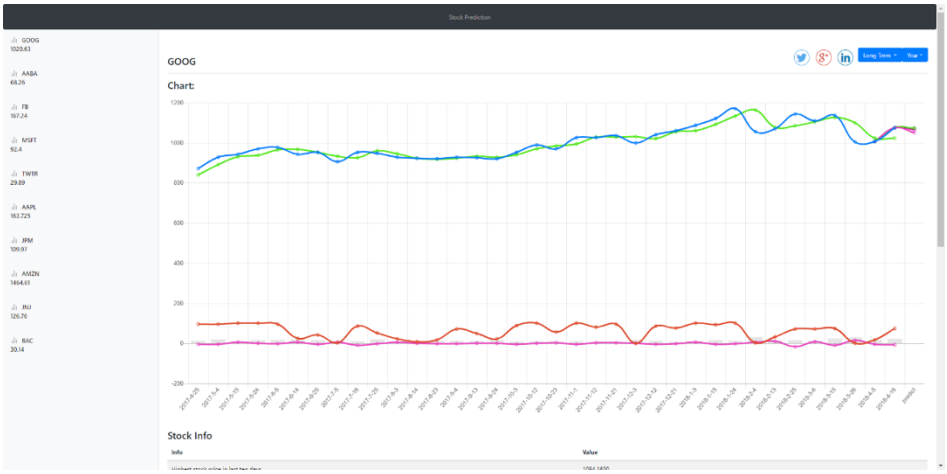
After we get the predicted price and the values of indicators, we can find out the final recommandation from these data. Each indicator will tell us that we should sell, buy or hold the stock, so we just simply let the indicators "vote" for the result, and we choose the decision that has the most votes. For example, if two or more indicators tell us to sell, then we recommand the user to sell the stock. And if all the three indictors tell us to do different things, then we recommand the user to hold the stock.

10 USER INTERFACE DESIGN AND IMPLEMENTATION

10.1 Welcome Page

This is welcome page of our RU RICH (web-based) system.

10.2 Home Page

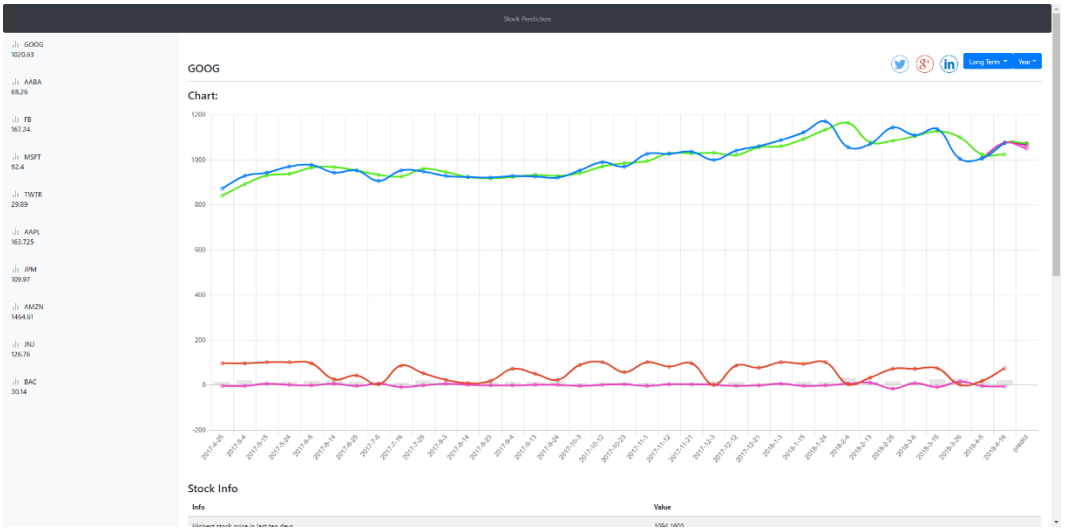


This is the home page of our system after the welcome page. After clicking

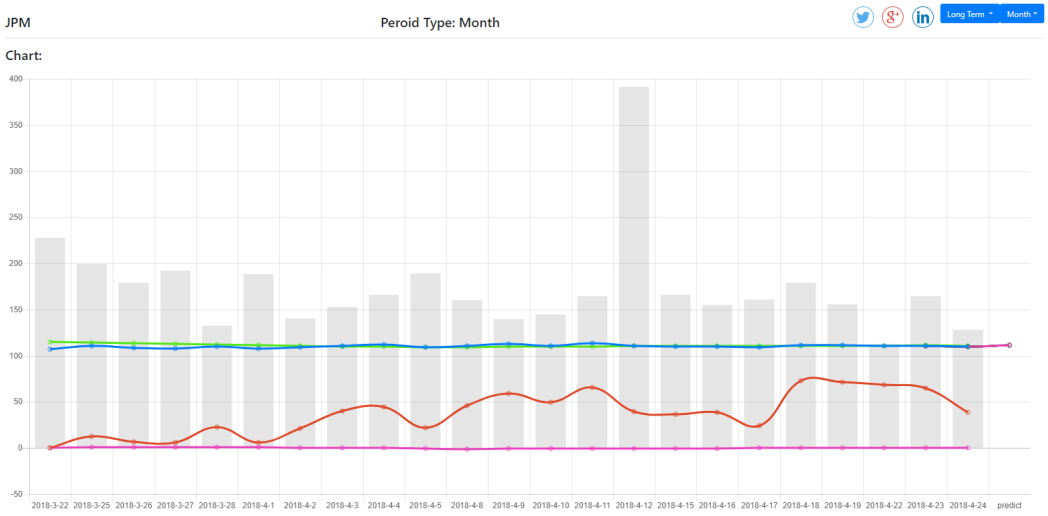
the FIND OUT MORE, you can find ten stocks listed in the left part.

10.3 Function Choose

Overview:



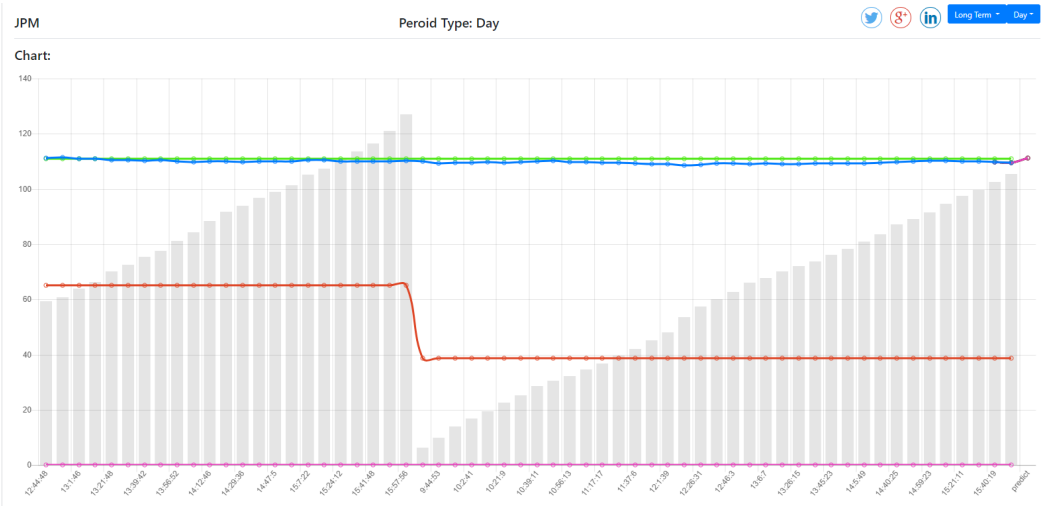
Historical Price:



You can choose a stock and choose the timescale of stock. The timescale

function button is on top right corner named Year, Month and Day.

Real Time Price:



You can choose a stock and choose the timescale of stock. You can see the real time stock price through choosing the day function. After that you will know the stock information and prediction strategy in the bottom of the real time stock price graph.

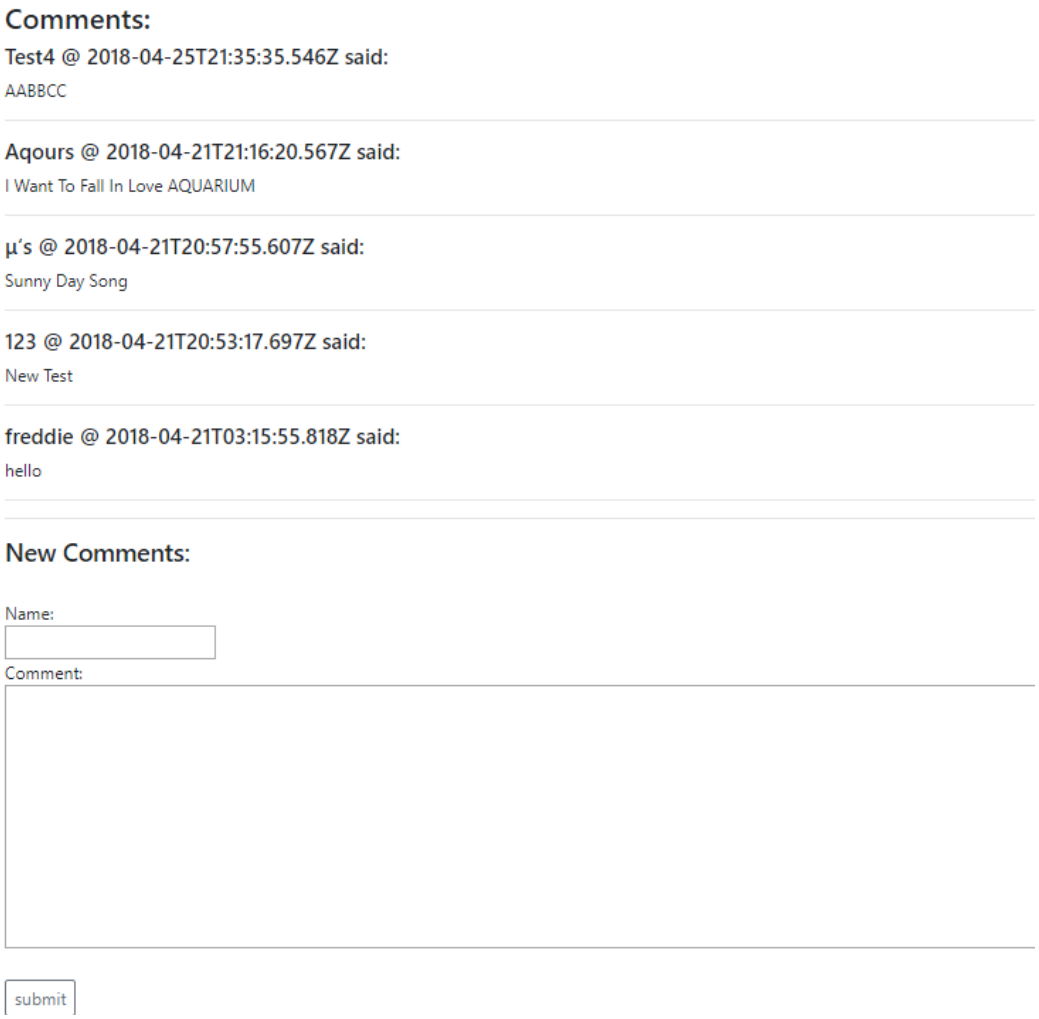
Stock Information:

Stock Info

Info	Value
Highest stock price in last ten days	1094.1600
Average stock price in last one year	1004.457896825396825396825396825397
Lowest stock price in last one year	867.7480
Stock with lower average price than selected stock	MSFT,AABA,AAPL,BAC,TWTR,JPM,JNJ,FB

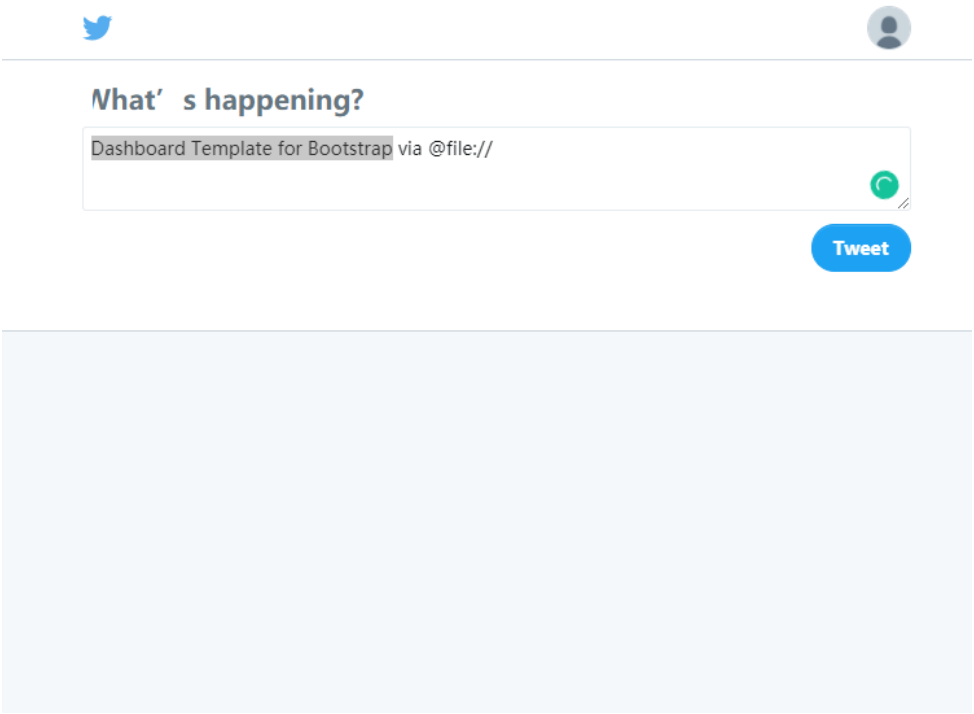
10.4 Special Function

Comment:



You can input your comment in the bottom input box. You need to fill in your name and comment and click the SUBMIT button to submit your comment.

Share:



You can share your comment and link to stock through Twitter, G+ and LinkedIn.

11 Related Work and Future Work

11.1 Related Work

There are many common used existing systems like Yahoo Finance, Google Finance, Bloomberg. They provide users to visit different stocks and view their summary. Also, users can query different types of data of the specific stock, offer the figure of the stock for a certain time period and predict the future stock price to help making investment decisions.

11.2 Future Work

First, we have to improve our front-end to provide a more user-friendly interface because our front-end is simple and crude now. The website will be more useful and easier to use.

Second, more functions will be added to the website. For example, users can search a certain stock by typing in its symbol into a search box. At the same time, login and sign up functions will be added, so we can send notification emails to users then.

At last, we hope to optimize our entire website architecture to decrease the CPU and memory used on the server.

12 Acknowledge

We would like to thank our Professor Shiyu Zhou and our graders Mahesh Hariharasubramanian and Fengru Li for their guidance and assistance.

13 Reference

[1]https://leibovitvrnewsletters.com/want-to-learn-more-about-my-vr-volume-reversal-indicator-watch-this-video/

[2]https://www.cs.ox.ac.uk/people/nando.defreitas/machinelearning/

[3]http://scikit-learn.org/stable/faq.html

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

[5]https://www.tensorflow.org/

[6]https://machinelearningmastery.com/a-gentle-introduction-to-scikit-learn-a-python-machine-learning-library/

[7]https://www.investopedia.com/ask/answers/031115/what-volatility-ratio-formula-and-how-it-calculated.asp

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

[9]http://stockcharts.com/school/doku.php?id=chart_school:technical_indicators:moving_average_convergence_divergence_macd

[10]http://stockcharts.com/school/doku.php?id=chart_school:technical_indicators:moving_averages

[11]https://www.alphavantage.co/

[12]https://en.m.wikipedia.org/wiki/Representational_state_transfer

[13]https://en.wikipedia.org/wiki/JQuery

[14] Software Engineering (online version), Chapters 6-8, by Ivan Marsic

[15] Web-Based Application Development, by Ralph F. Grove

Publisher: Jones & Bartlett Publishers; 1 edition (April 22, 2009)

ISBN-10: 0763759406 ; ISBN-13: 978-0763759407

[16] Pattern Recognition and Machine Learning, by Christopher M. Bishop, Springer, 2006.