SMART STOCK CONSULTANT

Team Members

Huang Xinrun , A0231555J
Guo Fangjie , A0231415U
Wang Yangxi , A0231505U

Aug-Oct 2021

Content

Executive Summary 2
1. Background 3
2. Commercial value 3
    2.1 Market analysis problem 3
    2.2 Solutions and advantages 4
3. Technical realization 4
    3.1 Knowledge acqusition 5
        3.1.1 Data sources 5
        3.1.2 Technical indicators 6
    3.2 Knowledge representation 7
        3.2.1 Decision tree 7
        3.2.2 Random forest 8
    3.3 Dialogflow agent 9
4. System design 10
    4.1 Overall architecture 10
    4.2 system detailed design 11
        4.2.1 User interface 11
        4.2.2 Algorithm implementation 12
        4.2.3 Prediction and recommendation module 13
        4.2.4 Chatbot module 16
    4.3 Result 19
5. Conclusion 21
Reference: 22
Appendix: 23
    Appendix-1: Project Proposal 23
    Appendix-2: Mapped System Functionalities 24
    Appendix-3: Installation and User Guide 26
    Appendix-4: Individual Project Report 34

Executive Summary

With the increasing social economy and better living conditions, more and more people choose stock investment as financial management. The investors' most concern of stock is how to predict the stock tendency correctly. There are many factors affecting stocks, such as companies' operating conditions, countries' policies and so on. Consequently, an efficient and precise stock prediction is one of hottest field today.

Nowadays, AI application is more and more extensive, applied to every bit of our life, and it begins to be applied to stock decision-making, overcoming the inaccuracy of traditionally mathematical statistics on companies' basic information. The inaccuracy of traditional method may be caused by delay of the company's information or national policy etc. While AI can learn and tune itself, what's more, it has the advantage of good fault tolerance compared with traditional methods.

Therefore, our team decide to build a machine learning model to predict the stock trend. First, we collect the historical stock data over 15 years from 30 companies ranked by its market capitalization. Then we calculate the 11 technical indicators as the features. After that we build the random forest model and fit the data to train our model. Our system has three main functions: prediction, recommendation and chatbot. Based on our trained model and current data, we would tell the user whether this company's stock will rise or not. Our system also recommends users the valuable stocks which is going to rise today and has potential to earn highest money. In addition, our system integrates with chatbot, making it available and easy for users to communicate with our system through Q&A chatbot, saving a lot of money on consulting on economical expert.

1. Background

Stocks, or shares of a company are financial instrument that represent ownership in a company or corporation and represents a proportionate claim on its assets and earnings. It is a kind of securities that stock holders use to obtain bonuses. Stockholders have two kinds of investment income. One is to distribute dividends according to the amount of shares held, and the other is to obtain income from changes in stock prices.

Nowadays, living conditions are getting better and better, and more and more people have the need to invest in financial management, which has caused more and more attention on stock investment in walks of life. Investors can gain profits in stock price changes by selling the transferred shares at any time. Therefore, the stock market has always been a topic of widespread concern. Many people's lives have changed because of the stock market. Famous successful case such as Warren Buffett, who is called the stock god. He earned the first pot of gold in his life through stocks, and he has successfully entered forbes list of billionaires through continuous stock investment.

2. Commercial value

2.1 Market analysis problem

The investors' most concern of stock is how to predict the stock tendency correctly. There are many factors affecting stocks, such as companies' operating conditions, countries' policies and so on. Consequently, an efficient and precise stock prediction is one of hottest field today.

First, in the current market, most people use traditional methods to make stock selections, such as using mathematical statistics on companies' basic information like macro-micro economy, company finance to make stock predictions. However, this type of method may not only be affected by delay of the company's information or national policy, but also is not accurate enough. Because in this way there are many different statistical indicators to measure a stock trend, and using different statistical indicators will get different results, which means we need to use one indicator or the combination of several indicators to make prediction, but this has more inaccuracy.

Second, to overcome the inaccuracy of traditional methods, some people may want to choose to invest stock by consulting the experts. But consulting the experts spends too much money and some experts do not have enough ability to predict stocks correctly, causing some people to be deceived.

2.2 Solutions and advantages

Nowadays, AI application is more and more extensive, applied to every bit of our life, and it can also be applied to stock decision-making. It uses machine learning methods to build models and make prediction. The system can learn and tune itself, what's more, it has the advantage of good fault tolerance compared with traditional methods. There are many methods to use AI to predict stock. For example, Guresen et al. used multi-layer contrast-aware (MLP), dynamic artificial neural networks and hybrid neural networks to predict Nasdaq stock data, which proved that hybrid neural networks have higher precision. Kimoto et al. used the stock data of the Tokyo Stock Exchange to construct a neural network-based TOPIX stock forecasting system for prediction. The prediction results showed that the neural network model has a higher rate of return. Thus, we want to apply machine learning model to make stock forecast. First, we collect the historical stock data over 15 years from 30 companies ranked by its market capitalization. Then we calculate the 11 technical indicators as the features. After that we build the random forest model and fit the data to train our model. Our system has two main functions: prediction, recommendation and chatbot. Based on our trained model and current data, we would tell the user whether this company's stock will rise or not. Our system also recommends users the valuable stocks which is going to rise today and has potential to earn highest money.

Overcoming the large cost and risk of consulting experts, we choose to add chatbot to our system, in order to achieve the goal of automatic question and answer. This can make our system easy to use and can save users' money. In our system, we get dataset from the stock exchange, and use random forests to make model for prediction. And we display the result on the website, users can also ask chatbot for stock choosing.

3. Technical realization

3.1 Knowledge acqusition

3.1.1 Data sources

1） Alpha Vantage (https://www.alphavantage.co/documentation/) – a website that provides enterprise-grade financial market data through a set of powerful and developer-friendly APIs.

The data we use includes date, daily open, daily high, daily low, daily close, daily volume, and the time span is 4000 trading days(about 16 years).

2） companiesmarketcap (https://companiesmarketcap.com/#) - a website that ranks the market value of global listed companies based on data from the New York Stock Exchange

We choose the Top30 companies by market capitalization, including Apple, Microsoft, Alphabet (Google), Amazon, Facebook, etc.

Figure 1 shows the trend of UnitedHealth's closing price in 4000 days. As shown in the figure, we can see that in 2008-2009 and 2019-2020, the close price of this company has experienced two significant declines. In the first time period, due to the global financial crisis, the stock price dropped from approximately $53 to $23, a 57% drop; while in the second time period, due to the impact of COVID-19 on the global financial market, the stock price of UHH decrease from almost $280 to $200, a drop of 29%.

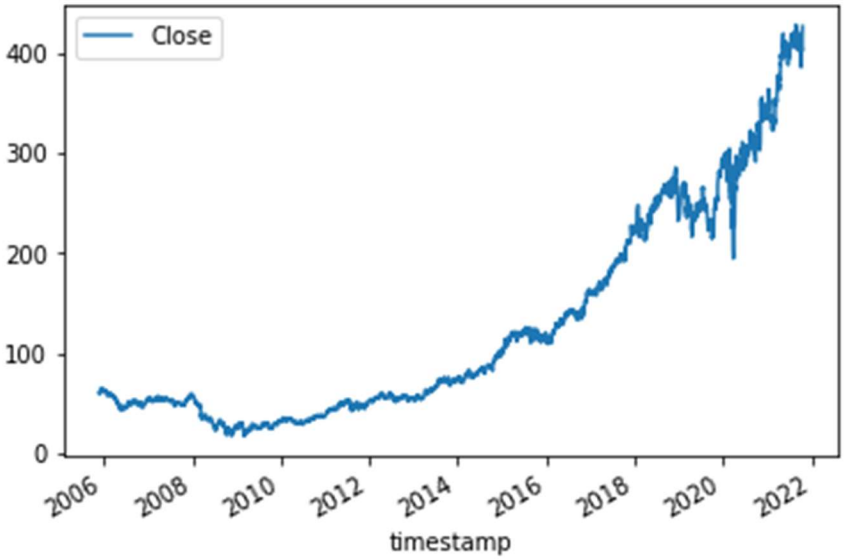

Figure 1. trend of UnitedHealth stock

3.1.2 Technical indicators

We calculated 11 indicators for model training and prediction, including MA, PSY, AROON_Up, AROON_Down, CCI, CMO, MACD, RSI, STOCHK, STOCHD and BIAS.

1）RSI (Relative Strength Index): describes a momentum indicator that measures the magnitude of recent price changes in order to evaluate overbought or oversold conditions in the price of a stock or other asset.

$$RSI = 100 - \frac{100}{1 + \frac{Average\ gain}{Average\ loss}}$$

2）MACD (Moving Average Convergence Divergence): a trend-following momentum indicator that shows the relationship between two moving averages of a security's price.

$$\text{MACD} = 12_{\text{Period}}\text{EMA} - 26_{\text{Period}}EMA$$

EMA: Exponential Moving Average

3）STOCH (Stochastic Oscillator): is a momentum indicator comparing a particular closing price of a security to a range of its prices over a certain period of time. The sensitivity of the oscillator to market movements is reducible by adjusting that time period or by taking a moving average of the result. It is used to generate overbought and oversold trading signals, utilizing a 0–100 bounded range of values.

$$\%K = \left(\frac{\text{C} - \text{L14}}{\text{H14} - \text{L14}}\right) \times 100\%$$

C = The most recent closing price

L14 = The lowest price traded of the 14 previous trading sessions

H14 = The highest price traded during the same 14-day period

%K = The current value of the stochastic indicator

4） CCI (Commodity Channel Index): a momentum-based oscillator used to help determine when an investment vehicle is reaching a condition of being overbought or oversold.

$$CCI = \frac{\text{Typical Price} - \text{Simple Moving Average}}{0.015 \times \text{Mean Deviation}}$$

Typical Price = asset's price on a particular day in the time period

Simple Moving Average = arithmetic mean of asset's price over a time period

Mean Deviation = mean of the absolute deviations of the asset's price over a time period

3.2 Knowledge representation

3.2.1 Decision tree

A decision tree is a map of the possible outcomes of a series of related choices. It allows an individual or organization to weigh possible actions against one another based on their costs, probabilities, and benefits. They can be used either to drive informal discussion or to map out an algorithm that predicts the best choice mathematically.

A decision tree typically starts with a single node, which branches into possible outcomes. Each of those outcomes leads to additional nodes, which branch off into other possibilities. This gives it a treelike shape.

There are three different types of nodes: chance nodes, decision nodes, and end nodes. A chance node, represented by a circle, shows the probabilities of certain results. A decision node, represented by a square, shows a decision to be made, and an end node shows the final outcome of a decision path.

Some criteria:

1） Entropy: a measure of the randomness in the information being processed

For 1 attribute:

$$E(S) = \sum_{i=1}^{c} -p_i \log_2 p_i$$

S: current state

Pi: probability of an event i of state S or Percentage of class i in a node of state S

For multiple attributes:

$$E(T, X) = \sum_{c \in X} P(c)E(c)$$

T: current state

X: selected attribute

2) Information Gain: a statistical property that measures how well a given attribute separates the training examples according to their target classification

$$Information\ Gain(T, X) = E(T) - E(T, X)$$

3) Gini index: a cost function used to evaluate splits in the dataset

$$Gini = 1 - \sum_{i=1}^{c} p_i$$

4) Reduction in Variance: an algorithm used for continuous target variables (regression problems). This algorithm uses the standard formula of variance to choose the best split. The split with lower variance is selected as the criteria to split the population.

$$\text{Variance} = \frac{\sum(X - \overline{X})}{n}$$

X-bar: the mean of the values

X: actual

n: the number of values

3.2.2 Random forest

A random forest is a machine learning technique that's used to solve regression and classification problems. It utilizes ensemble learning, which is a technique that combines many classifiers to provide solutions to complex problems.

A random forest algorithm consists of many decision trees. The 'forest' generated by the random forest algorithm is trained through bagging or bootstrap aggregating. Bagging is an ensemble meta-algorithm that improves the accuracy of machine learning algorithms.

The algorithm establishes the outcome based on the predictions of the decision trees. It predicts by taking the average or mean of the output from various trees. Increasing the number of trees increases the precision of the outcome.

3.3 Dialogflow agent

A Dialogflow agent is a virtual agent that handles conversations with your end-users. It is a natural language understanding module that understands the nuances of human language. Dialogflow translates end-user text or audio during a conversation to structured data that your apps and services can understand. The process of Dialog agent is as follow:

1) Intent recognition: An intent categorizes an end-user's intention for one conversation turn. When an end-user writes or says something, referred to as an end-user expression, Dialogflow matches the end-user expression to the best intent. Matching an intent is also known as intent classification. A basic intent contains the following:
Training phrases: These are example phrases for what end-users might say. When an end-user expression resembles one of these phrases, Dialogflow matches the intent. Dialogflow's built-in machine learning expands on defined list with other, similar phrases.

Action: When an intent is matched, Dialogflow provides the action to the system, and user can use the action to trigger certain actions defined in system.

Parameters: When an intent is matched at runtime, Dialogflow provides the extracted values from the end-user expression as parameters. Parameters are structured data that can easily be used to perform some logic or generate responses.

Responses: the defined text, speech, or visual responses to return to the end-user.

2) Entities:

Dialogflow provides predefined system entities that can match many common types of data. For example, there are system entities for matching dates, times, colors, email addresses, and so on. User can also create own custom entities for matching custom data. For example, a vegetable entity that can match the types of vegetables available for purchase with a grocery store agent.

3) Contexts:

Dialogflow contexts are similar to natural language context. If a person says "they are orange", you need context in order to understand what "they" is referring to. Similarly, for Dialogflow to handle an end-user expression like that, it needs to be provided with context in order to correctly match an intent.

Using contexts, user can control the flow of a conversation and configure contexts for an intent by setting input and output contexts, which are identified by string names. When an intent is matched, any configured output contexts for that intent become active. While any contexts are active, Dialogflow is more likely to match intents that are configured with input contexts that correspond to the currently active contexts.

4. System design

4.1 Overall architecture

As Figure 2 shows, our system uses the web application as our system interface since end-user can easily get access to our system with the web browser connected to network. And developer can be dedicated on system transactional development without maintain the client application. The front end of our system uses Jinjia2 Template, Semantic UI and JQuery techniques. And our back end applies the Flask Framework to handle the request from the front end, containing prediction, recommendation which use our trained model based on historical data and chatbot module. Each module design and procedure will be introduced afterwards. The chatbot module would use Google Dialogflow Platform to analyze users' intent and call Dialogflow API to request intent analysis and handle the intent in our Flask backend.

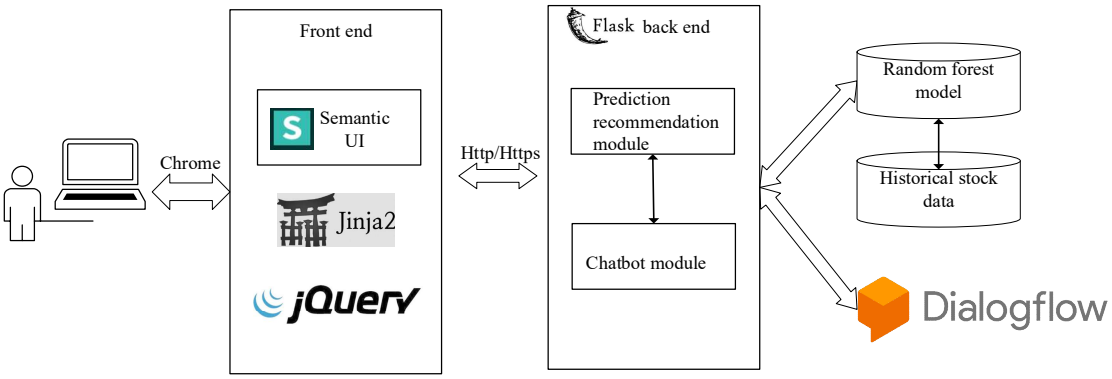

Figure 2. system architecture

4.2 system detailed design

4.2.1 User interface

The static web pages are designed as below. Figure 3 and 4 show the overview of basic company information and details of companies respectively in our system.

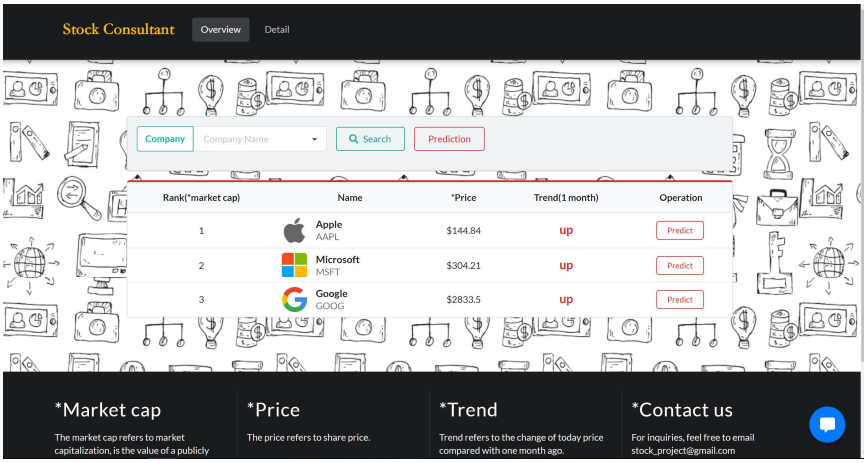

Figure 3. overview company page

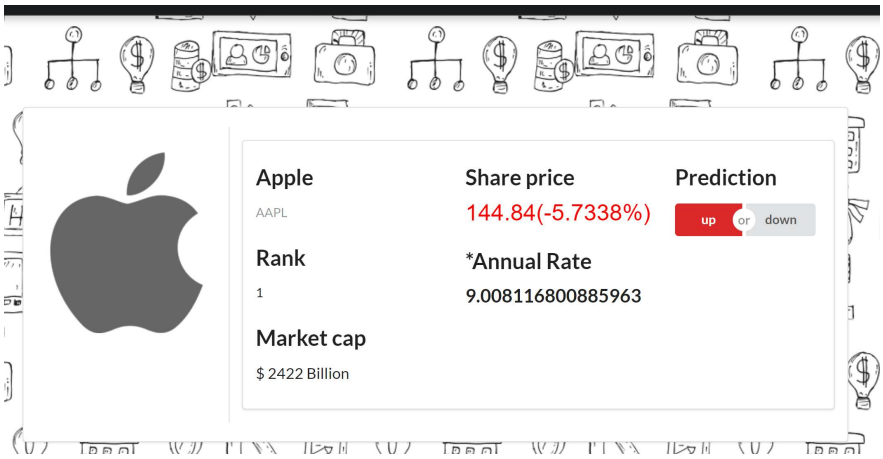

Figure 4. detailed company page

Figure 5 shows the chatbot UI design in the bottom right corner of our web page.

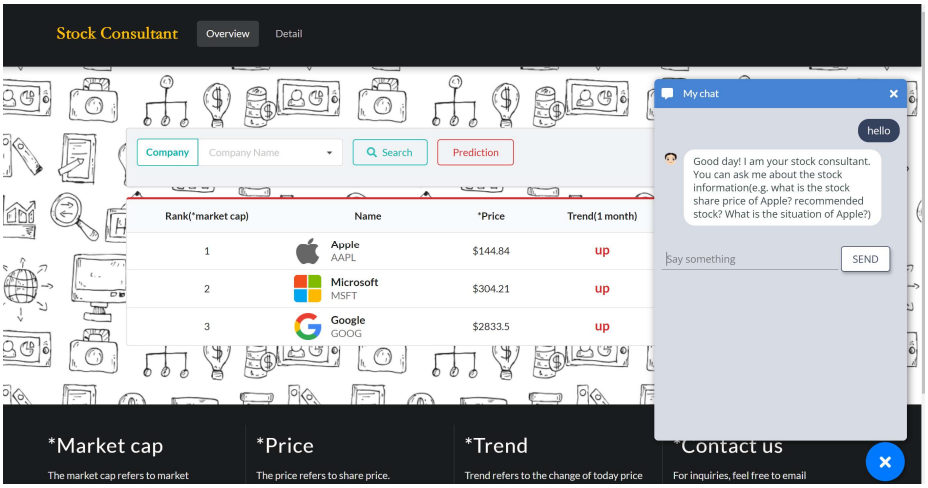

Figure 5. chatbot UI design

4.2.2 Algorithm implementation

The process of training the model is shown in the Figure 6 below. The detailed information are as follow.

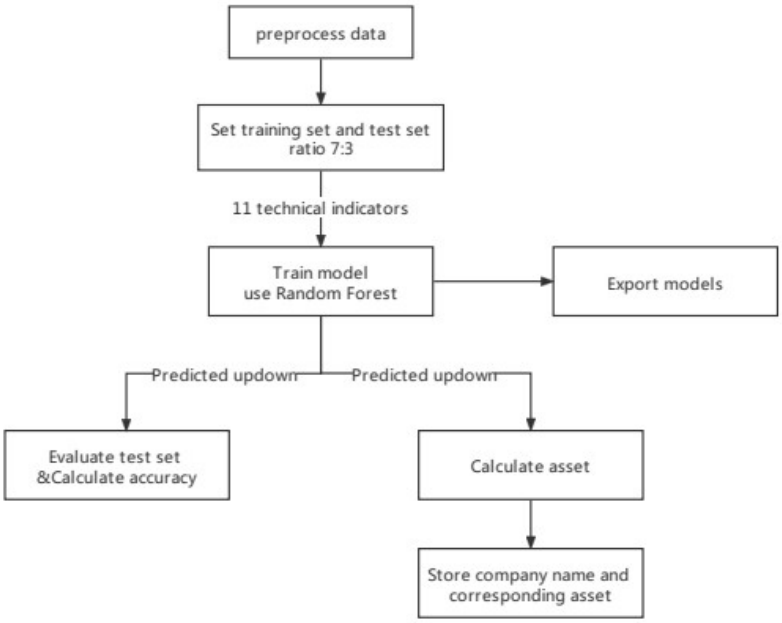

Figure 6. algorithm design process

1）preprocess data

As we analyze in knowledge acquisition chapter, we collect the historical stock price data in 4000 transactional days per company. And then we

1）Divide the data set

- The proportions of the training set and the test set are 70% and 30% respectively.

- x_test contains the 11 technical indicators mentioned above

2) Train Model

- Apply random forest model
- In order to improve the corresponding speed of the system and optimize its performance, we have trained the models of all these 30 companies in advance.

3) Export Model

Export the 30 trained models, which can be directly imported and invoked in the subsequent prediction process

4) Evaluation

- updown: Predicted result—whether the stock will rise that day
- asset: how many times the user's stock share on the day was the initial count
- Investment Strategy: If the stock price is predicted to fall on the day, no trading will be carried out; if the forecast is rising, buy with all the money and sell on the same day.
- Store company name and corresponding asset from May 2020 to May 2021

Test set evaluation

- Use the trained model to predict the last 30% of the data in the data set, and compare the results with the actual data. The accuracy rate can reach more than 90%.
- Comparative analysis indicators: accuracy, precision, recall, F1-score, figure 7 shows the results.

```
{'company': 'AAPL', 'accuracy': 0.91688, 'precision': 0.93333, 'recall': 0.91163, 'F1': 0.92235}
{'company': 'MSFT', 'accuracy': 0.92521, 'precision': 0.94761, 'recall': 0.91791, 'F1': 0.93252}
{'company': 'GOOG', 'accuracy': 0.95552, 'precision': 0.97161, 'recall': 0.95062, 'F1': 0.961}
{'company': 'AMZN', 'accuracy': 0.92527, 'precision': 0.94671, 'recall': 0.91654, 'F1': 0.93138}
{'company': 'FB', 'accuracy': 0.91013, 'precision': 0.8946, 'recall': 0.94054, 'F1': 0.917}
{'company': 'TSLA', 'accuracy': 0.9218, 'precision': 0.91703, 'recall': 0.9375, 'F1': 0.92715}
{'company': 'BRK-A', 'accuracy': 0.91842, 'precision': 0.9371, 'recall': 0.90923, 'F1': 0.92295}
{'company': 'TCEHY', 'accuracy': 0.93396, 'precision': 0.93004, 'recall': 0.93971, 'F1': 0.93485}
{'company': 'TSM', 'accuracy': 0.91765, 'precision': 0.93711, 'recall': 0.91131, 'F1': 0.92403}
{'company': 'NVDA', 'accuracy': 0.91604, 'precision': 0.93533, 'recall': 0.90951, 'F1': 0.92224}
{'company': 'V', 'accuracy': 0.92035, 'precision': 0.9344, 'recall': 0.92294, 'F1': 0.92863}
{'company': 'JPM', 'accuracy': 0.91772, 'precision': 0.92092, 'recall': 0.9179, 'F1': 0.91941}
{'company': 'BABA', 'accuracy': 0.93714, 'precision': 0.93333, 'recall': 0.94382, 'F1': 0.93855}
{'company': 'JNJ', 'accuracy': 0.9152, 'precision': 0.92892, 'recall': 0.90981, 'F1': 0.91926}
{'company': 'UNH', 'accuracy': 0.91849, 'precision': 0.9373, 'recall': 0.90952, 'F1': 0.9232}
{'company': 'SMSN.LON', 'accuracy': 0.92851, 'precision': 0.94617, 'recall': 0.91772, 'F1': 0.93173}
{'company': 'WMT', 'accuracy': 0.91681, 'precision': 0.92869, 'recall': 0.91242, 'F1': 0.92048}
{'company': 'BAC', 'accuracy': 0.92101, 'precision': 0.9248, 'recall': 0.9248, 'F1': 0.9248}
{'company': 'LVMUY', 'accuracy': 0.94544, 'precision': 0.96485, 'recall': 0.93526, 'F1': 0.94983}
{'company': 'HD', 'accuracy': 0.92947, 'precision': 0.93972, 'recall': 0.93109, 'F1': 0.93538}
```

Figure 7. model evaluation

4.2.3 Prediction and recommendation module

The prediction module obtains the company name selected by the user, downloads the

company's data for the day from the website, preprocesses and calculates the indicators, and calls the trained model to make predictions.

The recommendation module selects the companies that are predicted to rise according to the forecast results of 30 companies on the day, sorts their growth in one year, and selects the top five companies to recommend to users.

The detailed steps are as follows:

1）Choose a company

Get the company name that the user needs to query from UI

2）Download data

- According to the acquired company name, automatically downloaded the stock data set of the day from Alpha Vantage API
- Preprocess the data: Select the most recent 4000 days of data, and select all the data if the listing is less than 4000 days; then arrange the data in chronological order

3）Calculation indicators

Calculate 11 technical indicators of the stock on the day, such as MA, PSY, CCI, etc.

4）Prediction

- Import the corresponding trained model according to the company name
- Select 11 indicators of the date to be predicted and use the model to predict

5）Recommend

- Predict updown of all companies on the day. If the forecast result is an increase, record the company name and its asset.
- Sort by asset and find the TOP5 companies with the largest increase in a year

In the back-end, the functions designed as follow:

Table 1. function description

Function Name	description
setDF(path)	Read the original data, calculate 11 technical indicators and save them in a list
cul_accuracy_precision_recall(y_true, y_pred, pos_label=1)	Calculate accuracy, precision, recall and F1-score based on the predicted updown and actual updown
trainModel(df,company)	Train and export the corresponding random forest model according to the company name
predict_company(company_name)	Calculate the predicted updown based on the trained random forest model and company

	name
recommend_companies()	Recommend the Top5 profitable companies whose stock price would increase on the day

The flowchart of the prediction module is shown in the figure 8 below:

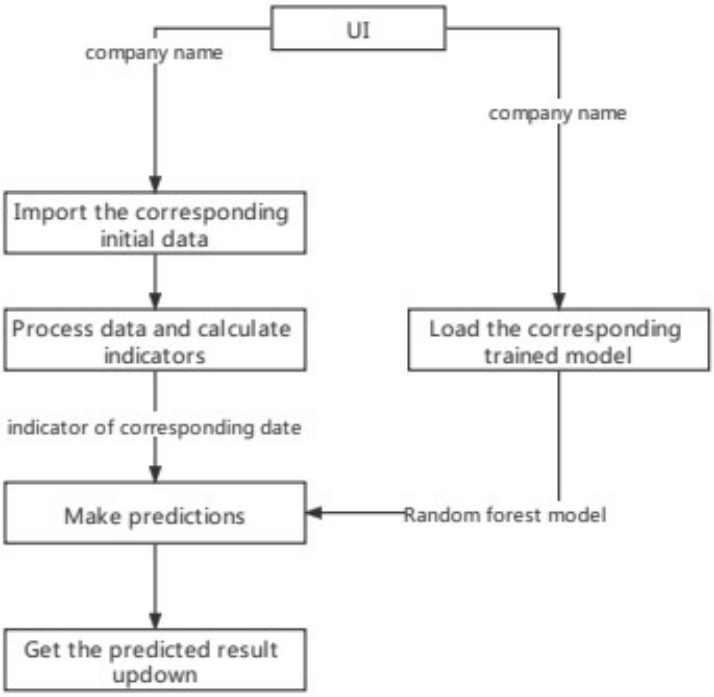

Figure 8. prediction module flowchart

The flowchart of the recommend module is shown in the Figure 9 below:

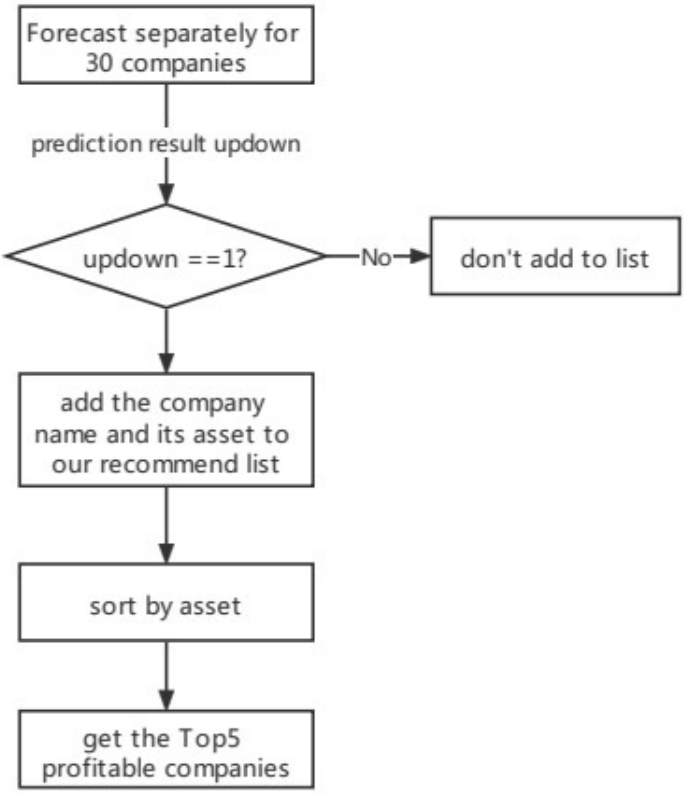

Figure 9. recommendation module flowchart

4.2.4 Chatbot module

In the chatbot module, we can achieve the task to ask the robot questions about the stock. In design, at first, we make some intents in the Dialogflow platform, then we send the intent which is analysed by the Dialogflow to the back-end. After that, we use the back-end design methods to answer different questions. Finally, we send the answer to the web page.

4.2.4.1 Intent introduction:

There are the intents designed in the Dialogflow as follow:

1) GetbestIntent:

Training phrases: Stock recommendation; What should I buy?

Default responses: None, try it again.

2) GetbuyIntent:

Training phrases: Should I buy Apple stock?; Can I buy the stock of Apple?

Default responses: None, try again.

3) GetHelpIntent:

Training phrases: Hello; Help

Default responses: Hello, I am your stock consultant. You can ask me about the stock information(e.g. what is the stock share price of Apple? what is the situation of Apple?)

4) GetStockIntent:

Training phrases: what is the stock share price of Apple?; what is the stock price of Apple?

Default responses: None, try again.

5) GetSupport:

Training phrases: Supported company.

Default responses: We support all companies in this page like Apple,Microsoft,Google,Amazon,Facebook,Tesla,BerkshireHathaway,Tencent,TSMC,NVIDIA,Visa,JPMorganChase,Alibaba,Johnson&Johnson,UnitedHealth,Samsung,Walmart,BankofAmerica,LVMH,HomeDepot,KweichowMouta,Mastercard, Procter&Gamble,Nestle,Roche,ASML,Walt Disney,Paypal,Adobe,Salesforce.

6) GetupordownIntent:

Training phrases: What is the prediction of Apple stock today ?; What is the situation of Apple?

Default responses: None, try it again

4.2.4.2 Processing dialog example

In this part, we use an example to explain the steps of processing. We assume that the user input is ‘Can I buy the stock of Apple’. The figure 10 shows the details.

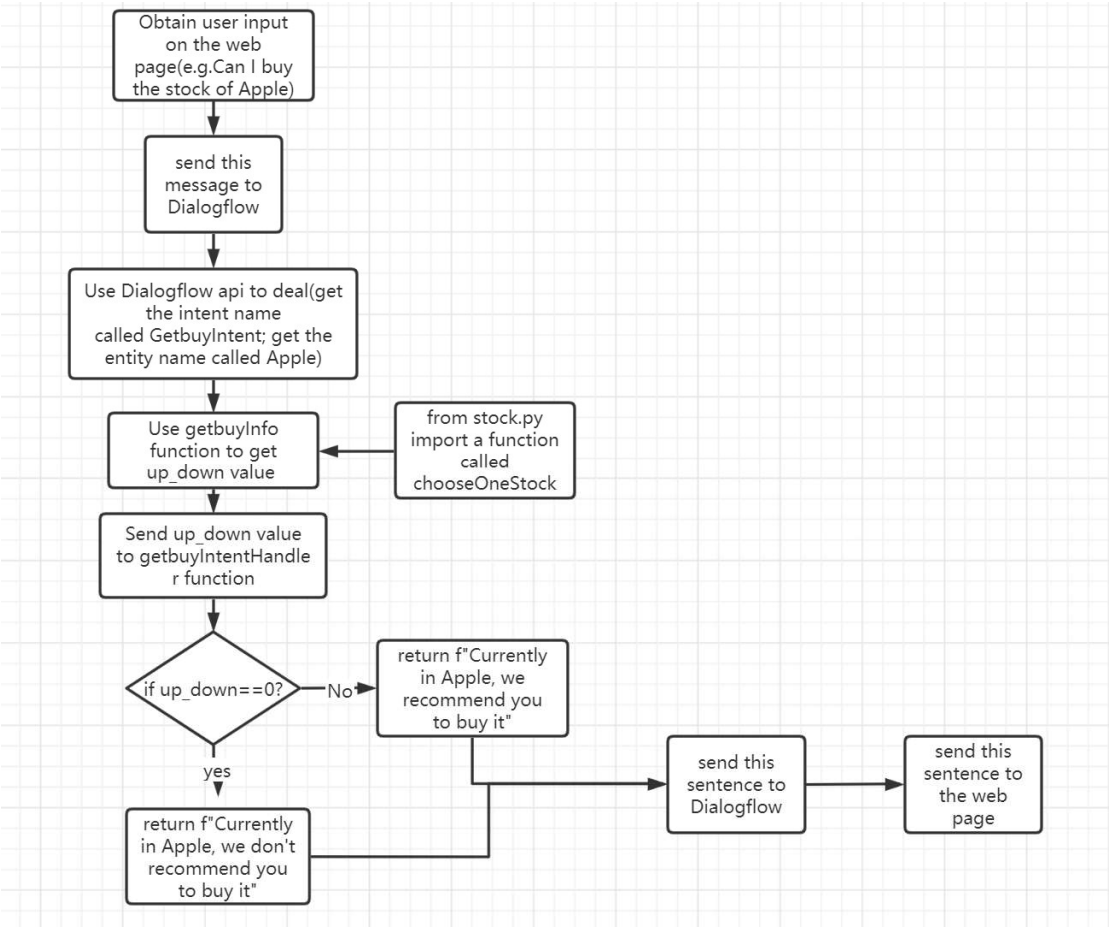

Figure 10.dialog processing

4.2.4.3 function summary

In the back-end, the functions designed as table 2 follow:

Table 2. function summary

Function name	description	input	output
getStockInfo	Get the stock price of a specific company.	A company name	Price value .
getupordownInfo	Get the stock price trend of a specific company.	A company name	A binary value which means trend.
getbuyInfo	Get the prediction of a specific company's stock price.	A company name	A array which contain the prediction.
getStockIntentHandler	Get the stock price from getStockInfo function, and return the sentence.	A company	A answer sentence.
getupordownIntentHandler	Get the binary value from the	A company name	Two different answer sentences

	getupordownInfo function and judge the value , return two different sentences.		according to binary values.
getbuyIntentHandler	Get the array from the getbuyInfo function, and get the prediction binary value, return two different sentences.	A company name	Two different answer sentences according to binary values.
webhook	Communicate with the web page, perform the intent analysis, and then call the corresponding function to get the result	none	Return the answer squences.

4.3 Result

After we design the user interface and various module, the running result of our system are shown below.

Figure 11 shows the overview of our supported 30 companies.

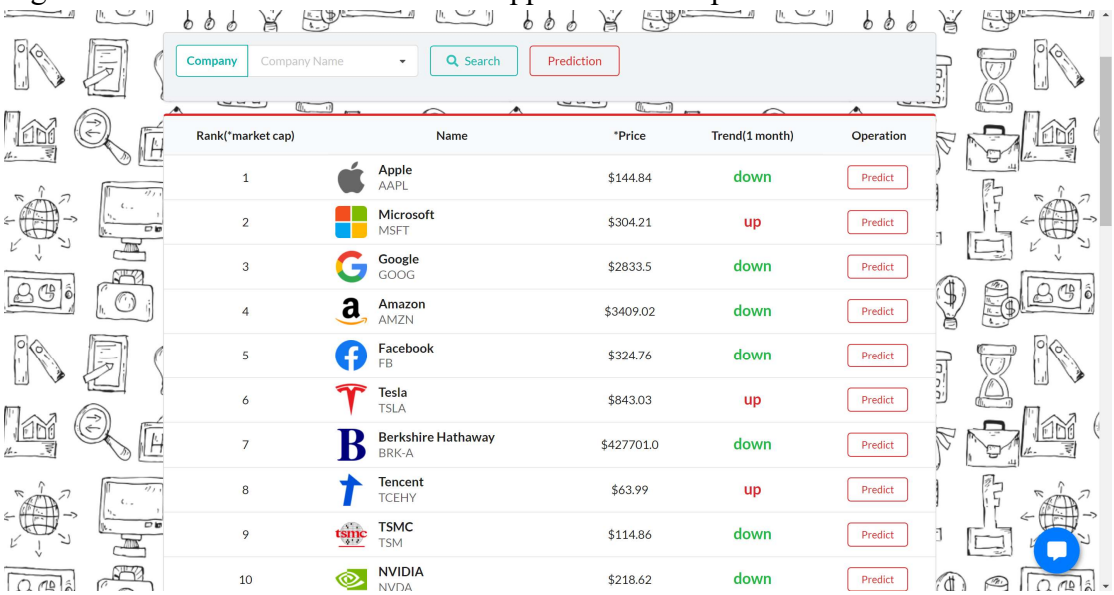

Figure 11. system overview

Figure12 shows the result of searching interested companies.

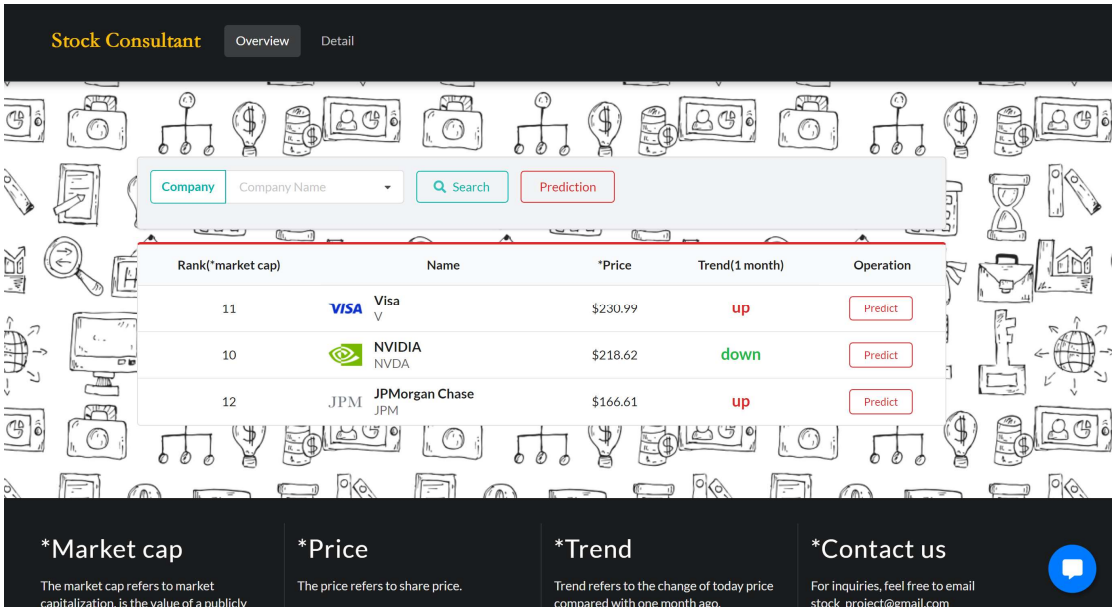

Figure 12. searching result page

Figure 13 shows the prediction of specific companies.

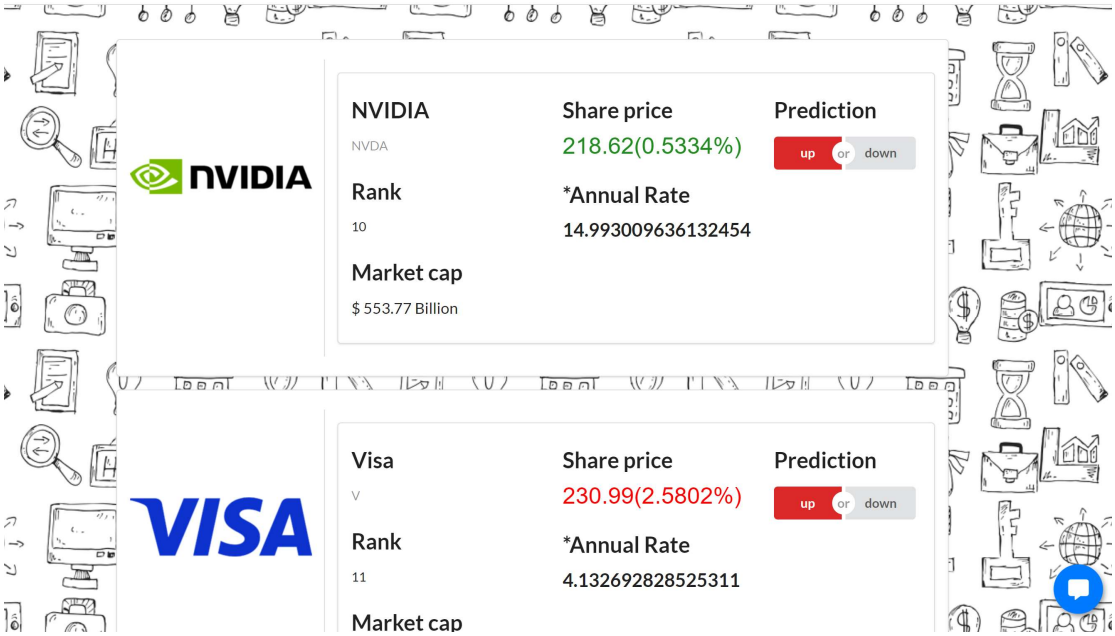

Figure 13. prediction result

Figure 14 and 15 shows the chatbot result with the question about 'should I buy apple stock' and 'recommended company' respectively

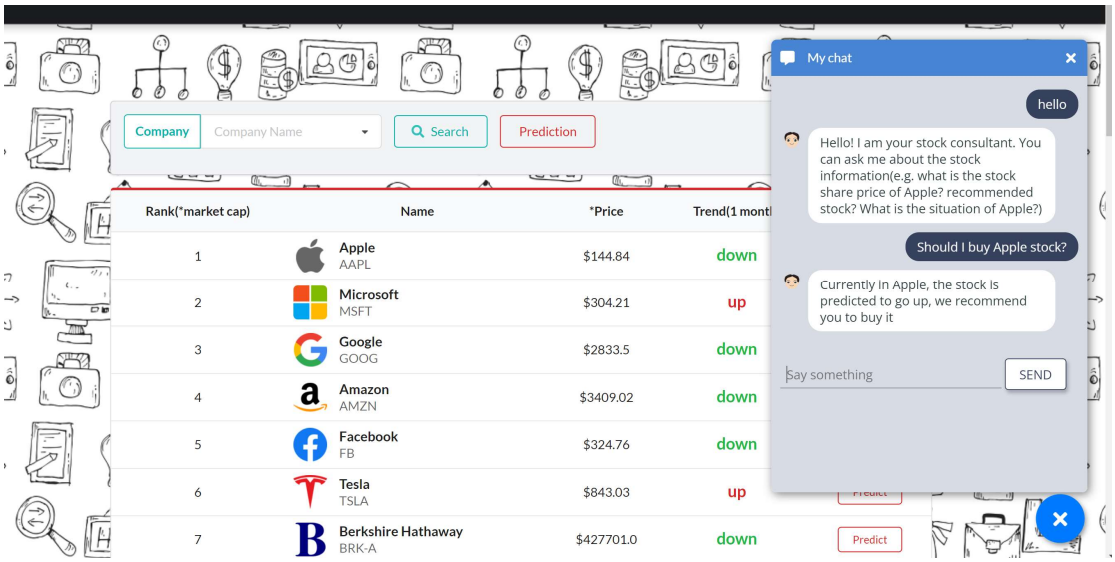

Figure 14. prediction result

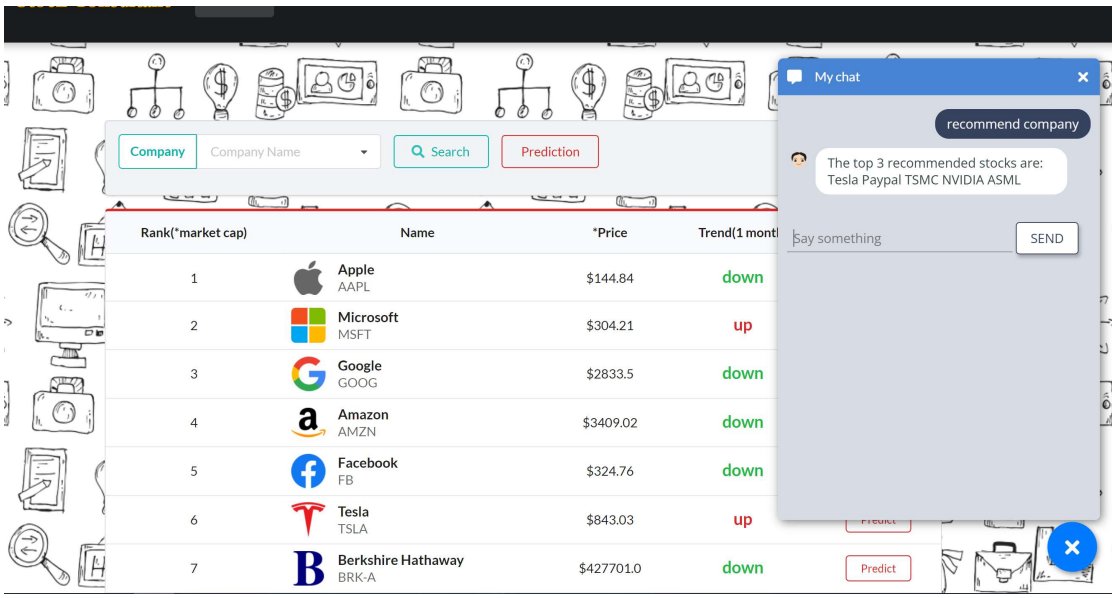

Figure 15. recommendation result

5. Conclusion

In this project, we build an Intelligent Stock Consultant system to predict the movement of stock integrated with Chatbot which automatically answer the question of investors. The result shows our model used in stock prediction is feasible with accuracy over 90%. And our system has a very user-friendly interface which means end-users can use it easily even without instructions. Therefore, our system has highly commercial values which can satisfy people's investment needs and provide accurate prediction compared

with other approaches. In addition, people can get access to the system freely through the Internet without spending tremendous time and money consulting experts.

In general, our team build the system with the combination of the learned knowledge in our courses, like knowledge representation, data mining techniques used in processing historical stock data and building random forest model, search and optimization techniques for recommending top 3 highest return stock companies, cognitive system and reasoning for understanding users' intent and generate answer. After we finished our project, we have better understanding of these knowledge. And we learn how to analyzing practical problem and make our knowledge practical.

Reference:

[1] Predicting Stock Prices Using Machine Learning
https://neptune.ai/blog/predicting-stock-prices-using-machine-learning
[2]Hiransha, M., et al. "NSE stock market prediction using deep-learning models." Procedia computer science 132 (2018): 1351-1362.
[3] Alpha Vantage. Stock historical data
https://www.alphavantage.co/documentation/
[4] companies' market capitalization
https://companiesmarketcap.com/#
[5] Shah, Vatsal H. "Machine learning techniques for stock prediction." Foundations of Machine Learning| Spring 1.1 (2007): 6-12.

Appendix:

Appendix-1: Project Proposal

Date of proposal: August 25, 2021
Project Title: Smart stock Prediction
Background/Aims/Objectives: Nowadays, living conditions are getting better and better, and more and more people have the need to invest in financial management, which has caused more and more attention on stock investment in walks of life .But in the current market, most people use traditional methods to make stock selections, which is not accurate enough. Other people may choose to consult the experts, which spends too much money and may be deceived. We aim to develop a system that can get dataset from the stock exchange, and use decision tree and random forests to make model for prediction. Plus, we choose to add chatbot to our system. In this way, our system is more accurate and easier to use.
Project Descriptions: The stock prediction problem solver mainly has two tasks: stock prediction and chatbot. Prediction and recommend module should make predictions according to the company's data for the day from the website and recommend . The procedure of this module contains that: 1) Data preparation: Select the top 30 companies in the New York Stock Exchange by market capitalization, according to these company names, automatically downloaded the stock data set of the day from Alpha Vantage API, and select the most recent 4000 days of data, and select all the data if the listing is less than 4000 days; then arrange the data in chronological order. The proportions of the training set and the test set are 70% and 30% respectively. The x_test contains the 11 technical indicators which can evaluate stock. 2) Model training: Apply random forest model. In order to improve the corresponding speed of the system and optimize its performance, we have trained the models of all these 30 companies in advance. 3) Evaluation: Evaluate the trained model performance with the precision and so on. 4) Export model: Export the 30 trained models, which can be directly imported and invoked in the subsequent prediction process. 5) Input: The company name that the user needs to query from UI

6) Prediction and recommend: Import the corresponding trained model according to the company name. Select 11 indicators of the date to be predicted and use the model to predict. According to the prediction, we can do recommendations.

The chatbot module can chat with the user and return the response which gets from the prediction and recommend module to the user.

1) Data collection: Obtain user input on the web page.
2) Intent analysis: Use Dialogflow api to deal with the input sentences and get the intent.
3) Function call: According to the intent, use the corresponding function to get the respond.
4) Respond representation: Send the respond to the Dialogflow, then send it to the web page to answer the user.

Appendix-2: Mapped System Functionalities

1. machine reasoning system

1) Perceive: we choose the Top30 companies by market capitalization, including Apple, Microsoft, Alphabet (Google), Amazon, Facebook, etc. And the data we use includes date, daily open, daily high, daily low, daily close, daily volume, and the time span is 4000 trading days (about 16 years).
2) Recognize: we calculated 11 indicators for model training and prediction, including MA, PSY, AROON_Up, AROON_Down, CCI, CMO, MACD, RSI, STOCHK, STOCHD and BIAS.
3) Learn: we apply random forest model to train the models of all these 30 companies in advance.
4) Reason: we use the trained model to predict the last 30% of the data in the data set, and compare the results with the actual data. The accuracy rate can reach more than 90%.
5) Act: give a forecast value based on the company name selected by the user and select the top five companies that are expected to rise and recommend them to users

2. AI framework

We get the company names that the user needs to query from UI. If there are multiple company names, it will be broken down into multiple sub-tasks to query, each query for a single company.

Then determine whether there is a model of this company that has trained.

If not, we need to train the model first. We divide the training set and the test set at a ratio of 7:3. Apple random forest model to train the training set. And export the trained model, which can be directly imported and invoked in the subsequent prediction process.

If yes, we need to determine whether there is data for the company that day. If not, our system automatically downloads the stock data set of the day from Alpha Vantage API, and calculates 11 technical indicators. The next is the prediction and recommend module. We import the corresponding trained model and select 11 indicators to do it.

3. knowledge Discovery using Data Mining Models

We collect stock data from Alpha Vantage API, which includes date, daily open, daily high, daily low, daily close, daily volume, and the time span we set is 4000 trading days (about 16 years). Then our system calculates the 11 technical indicators of the stock on the day, such as MA, PSY, CCI, etc. After that, we apply random forest model to train and predict these indicators.

4. Informed search optimization

Hill climbing (greedy best first search)

Predict whether the stock price will rise of all companies on the day. If the forecast result is an increase, record the company name and its growth rate.

Sort by growth rate and find the TOP5 companies with the largest increase in a year

5. cognitive techniques or tools using NATURAL LANGUAGE COMPREHENSION AND PROCESSING

In the chatbot module, we can achieve the task to ask the robot questions about the stock. In design, at first, we make some intents in the Dialogflow platform, then we send the intent which is analysed by the Dialogflow to the back-end. After that, we use the back-end design methods to answer different questions. Finally, we send the answer to the web page.

Appendix-3: Installation and User Guide

Overview

This document will introduce how to install our system and instruct users how to use our system. Our stock consultant system consists of 2 modules: prediction, recommendation and chatbot module.

For the prediction and recommendation module, user can predict the stock tendency today according to interest of companies as well as get recommendation stock lists according to the predicting trend. As for chatbot module, user can get information like (overview or prediction of the company, recommendation stock) through typing questions on the chatbot window. Our system would answer the questions automatically.

System requirements

1. Operating system: windows 10
2. Anaconda or new Python environment.
3. An account successfully sign up in Google Account.
4. A modern web browser Chrome version 94 or above.

Installation

1. Download the project

Down load the full project through google drive:
https://drive.google.com/file/d/1ep2sh_dNNtpIdCGuZ8vTukZm9RFxO2me/view?usp=sharing

unzip and cd to project root directory <your download path> /stock/

2. set up environment

- Open the cmd console use the Anaconda to create a new Python environment through commend:

```
conda create -n stock python=3.6
```

- Activate the new environment:

```
conda activate stock
```

- Install the required packages:

```
pip install -r requirements.txt
pip install TA_Lib-0.4.19-cp36-cp36m-win_amd64.whl
```

- Train our system model by:

```
python training.py
```

- In the current directory of File Explorer, press the SHIFT and click the mouse right button to enter the Powershell console. Enter the command:

```
Ngrok.exe http 5000
```

You will see the result below:

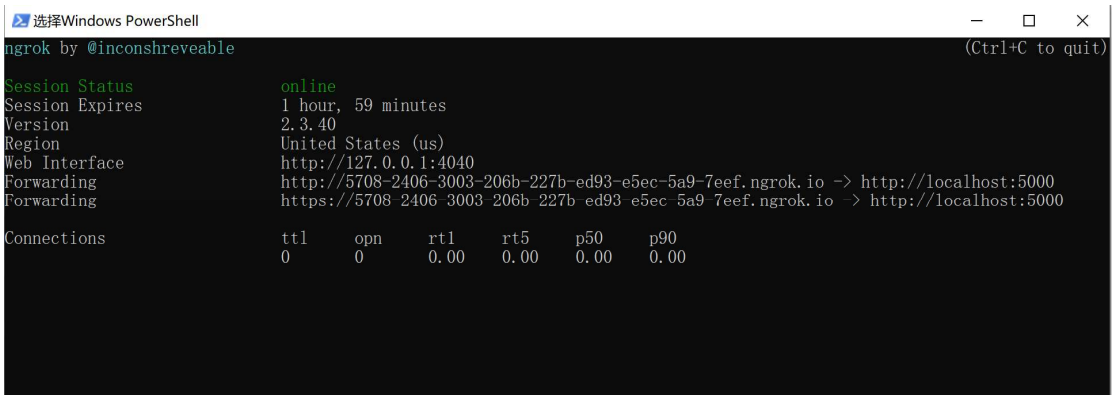

Figure 1. run the nerok.exe

- enter the Google Dialogflow throw link: https://dialogflow.cloud.google.com/ , sign in and create a new agent shown as Figure 2. Name the new agent as Stock.

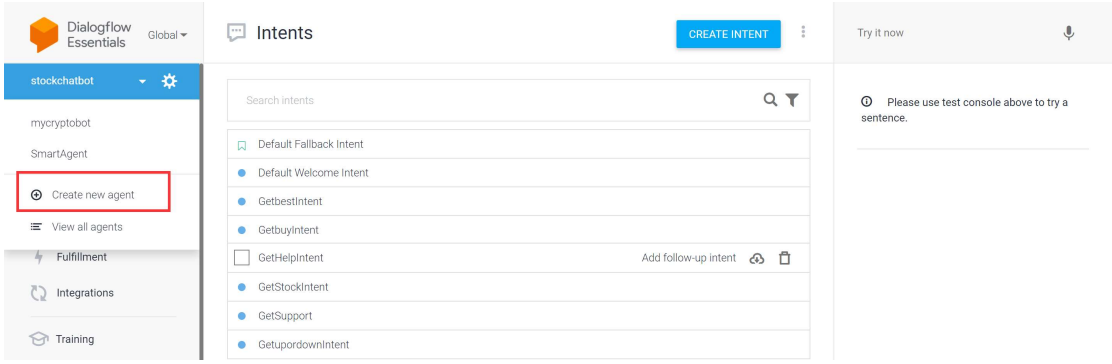

Figure 2. create a new agent

- Export the intent materials of our systems, click the gear icon and click the Export and Import, then click the Import from zip button to import intent material named stockchatbot.zip in our project root directory.

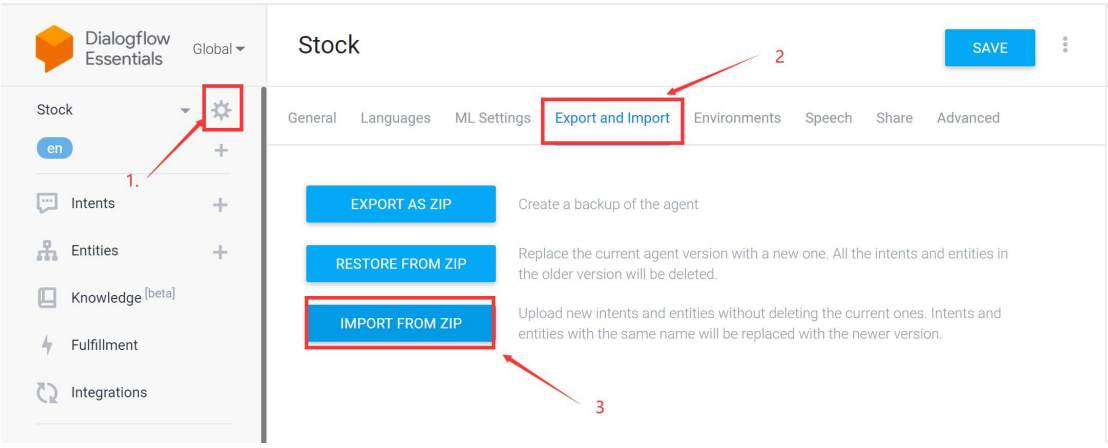

Figure 3. import the agent material

- Find the forward link in Figure 1 and click the Fulfillment in the left menu, then copy the forward link appending the '/google' in the end on URL* Input. The process is shown in Figure 4. Remember to click save in the end.

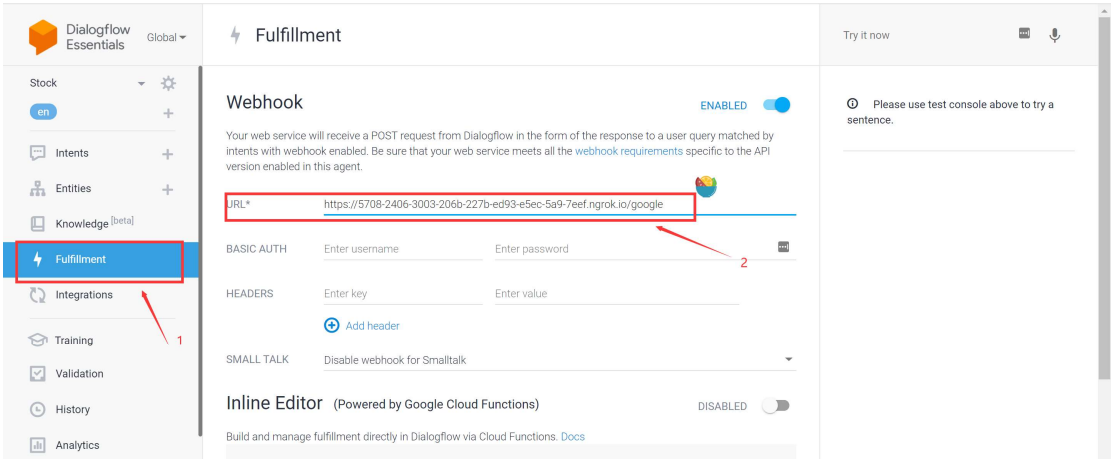

Figure 4. enter the fulfillment backend link

- Run the server.py script to launch our project backend through commend:

  `python server.py`

  The result will shown in Figure 5 if it is running successfully.

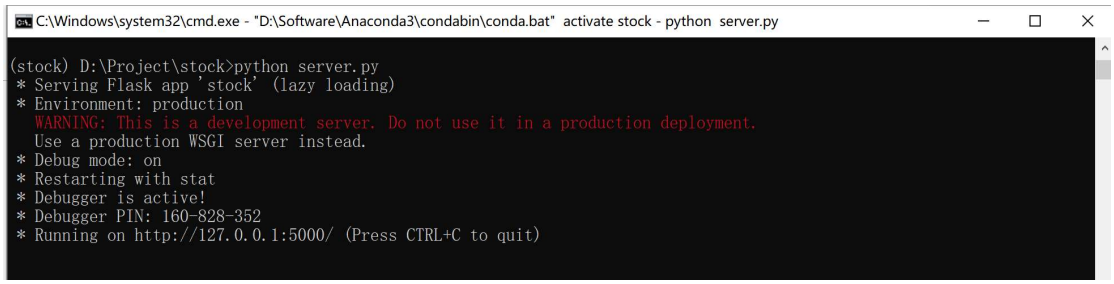

Figure 5. run server.py

- Launch the Chrome browser and type the link: http://localhost:5000/

User guide

Launch the project successfully and go to the URL http://localhost:5000/ , users can interacte with system through web UI or system chatbot.

- **Web UI**

The index page show the basic information of all 30 companies shown as Figure 6.

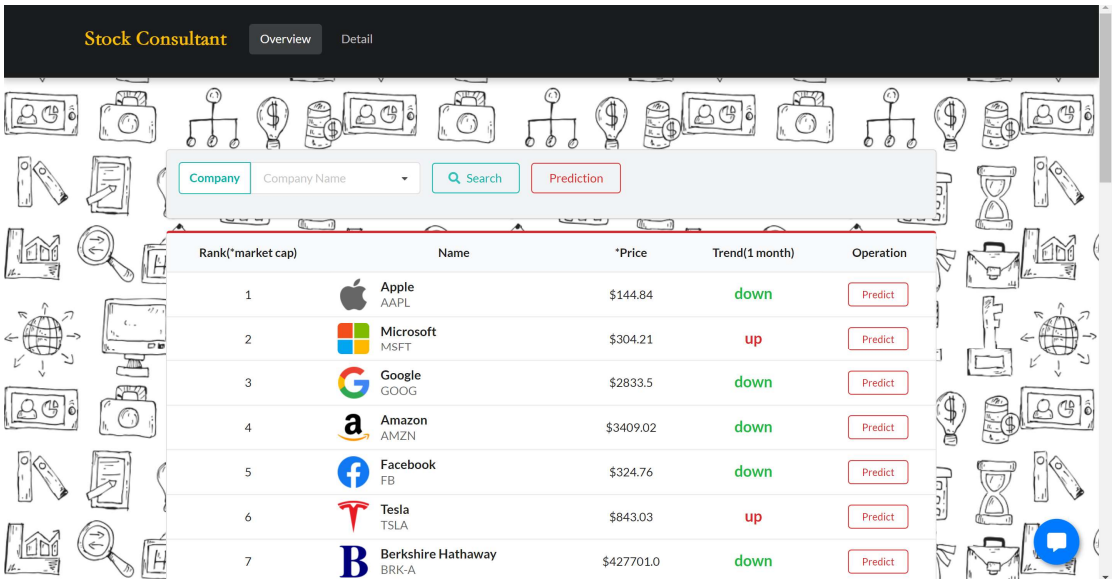

Figure 6. index page

30 Companies are ranked by their market capitalization, the information contains the company name, stock symbol, stock price and trend in one month.

Click the red prediction at the end of one row, our system will redirect to new page consisting of detailed and prediction information shown as Figure 7.

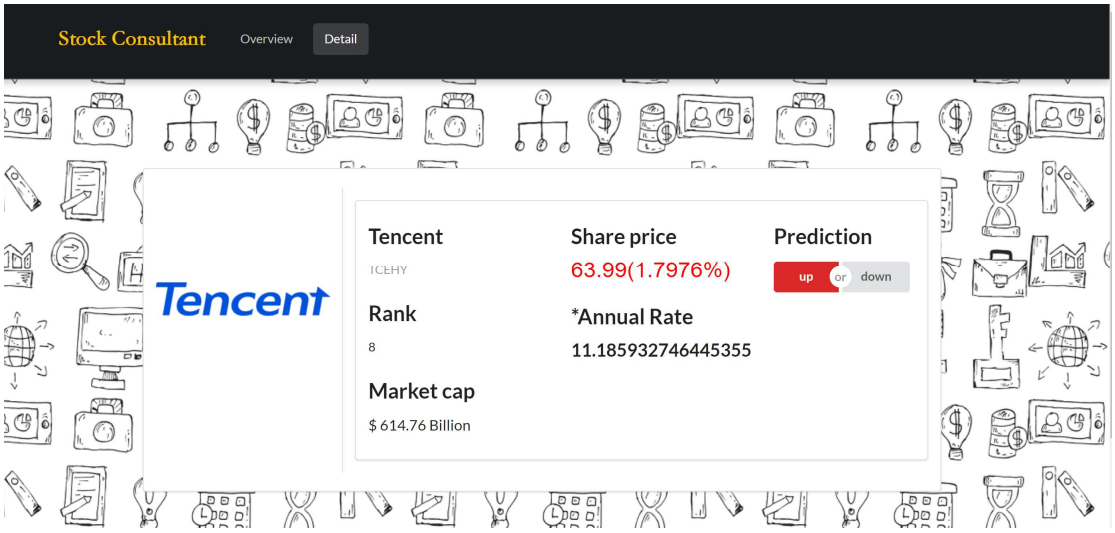

Figure 7. prediction inforamtion

The detailed information is made of company's name, symbol, market capticalization, share price, annual rate (The annual return of stock investment if we apply our machine learning method on investment.) and prediction information

On the other hand, user can search several companies of their interests in the segment above company's table in index page. It is a selection input which user can use to select several options or user can type company name and our system can automatically display the companies that are similar to user's input, shown as Figure 8 and Figure 9.

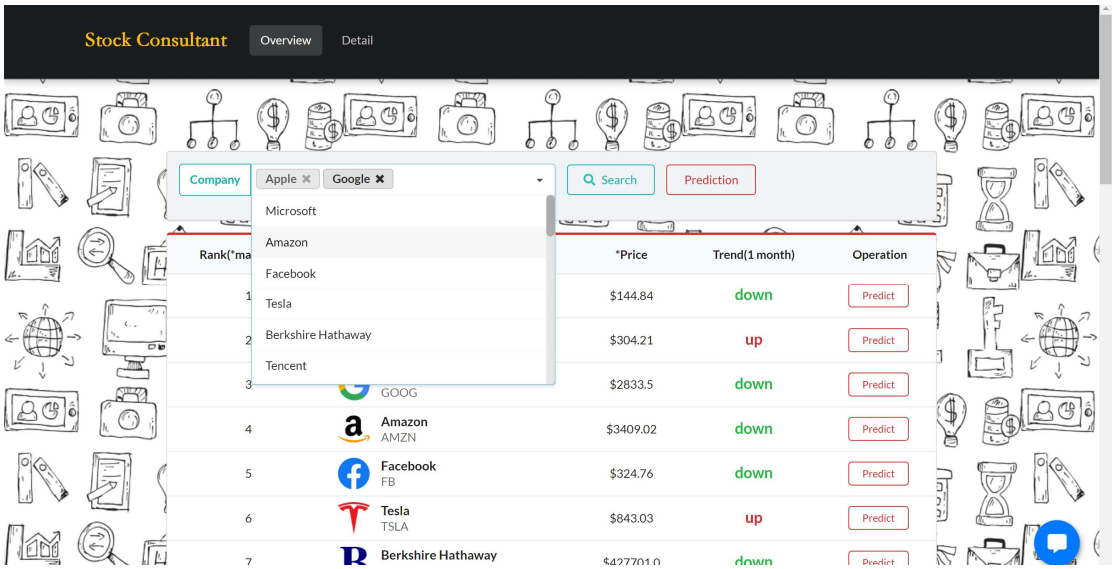

Figure 8. selection input

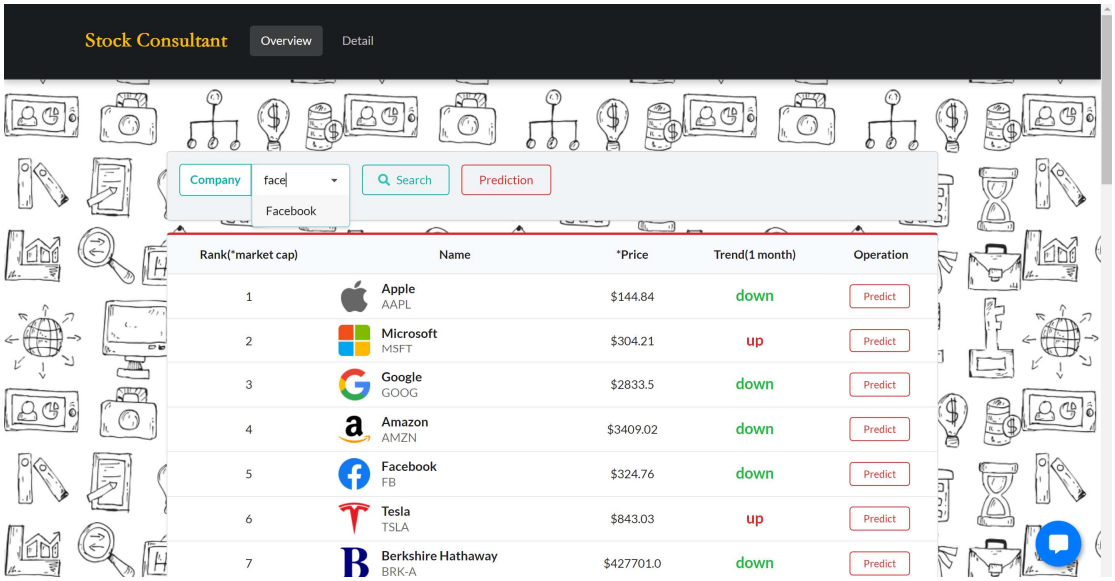

Figure 9. display the companies similar to input

Click the search button, it would return the specific companies to users, which is shown in Figure 10. Or users can click prediction button, the system will redirect to the detail information of selected companies, which is shown in Figure 11.

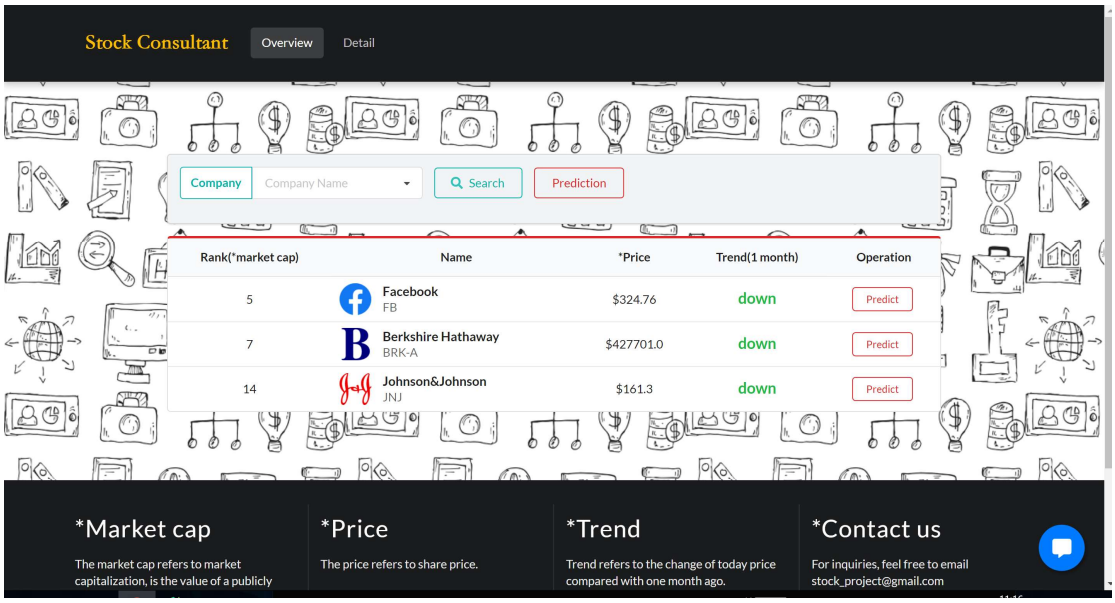

Figure 10. Search result

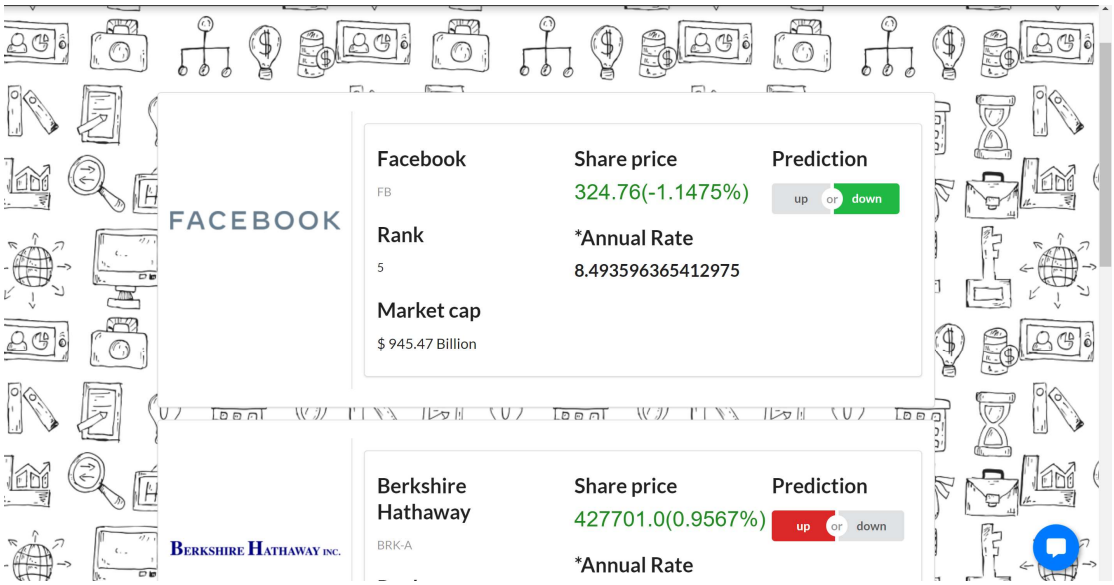

Figure 11. result of prediction for specific companies

- **Chatbot**

User can also interact with our system through chatbot window in bottom right of interface, which is shown in Figure 12.

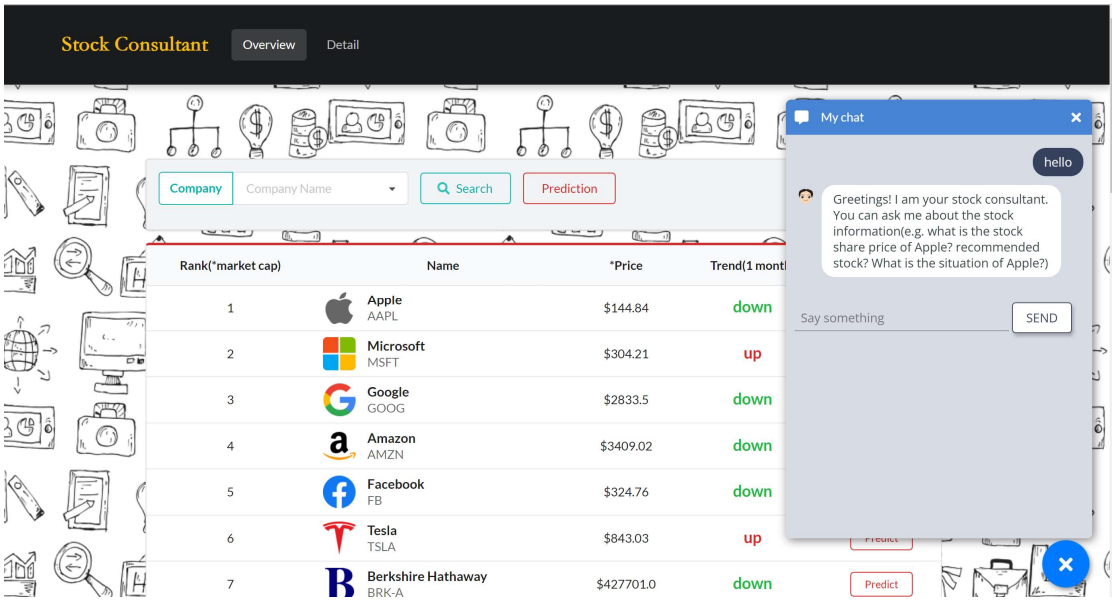

Figure 12. smart stock consultant chat window

Enter the question 'what is the stock share price of Apple?', our smart stock consultant will answer the question automatically. The result is shown in Figure 13.

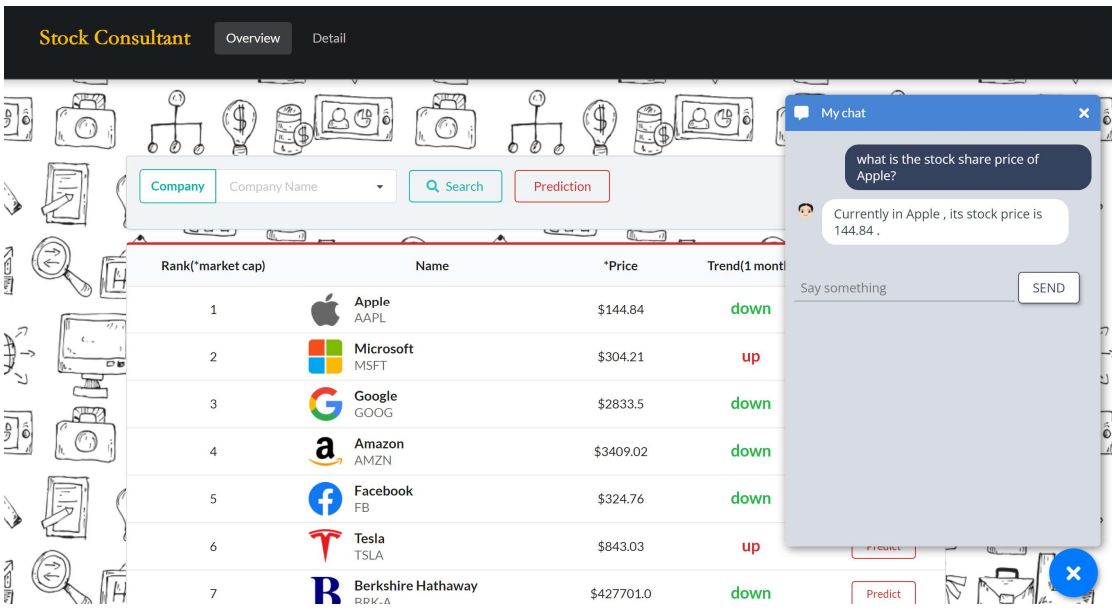

Figure 13. result of querying stock prediction

Figure 14 shows the result of recommendation question.

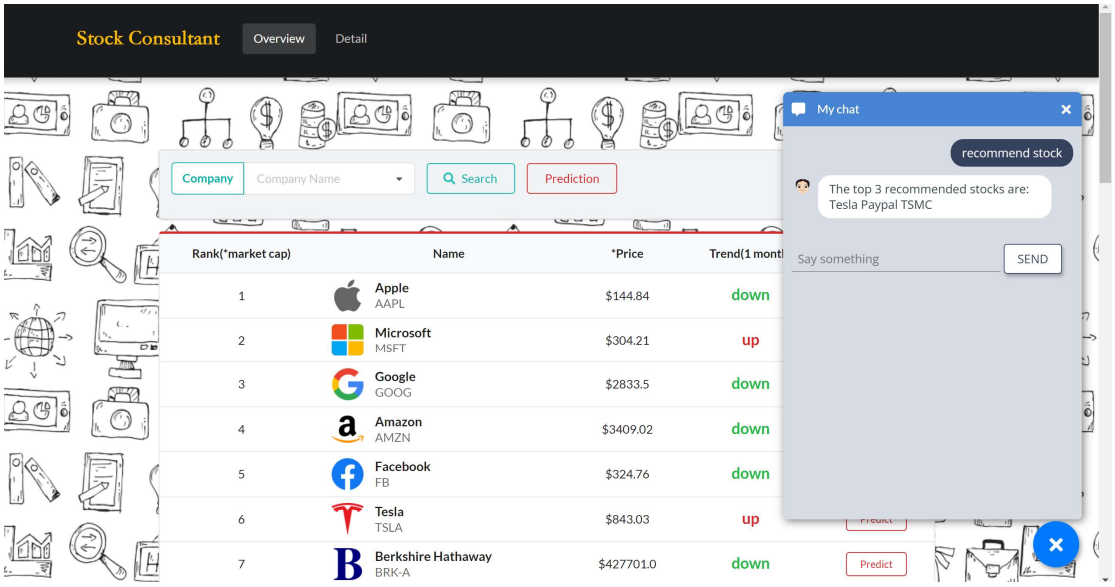

Figure 14. result of querying recommendation stock

Our supported questions are listed in Table 1.

Table 1. supported questions

type	question	answer
help	Hello, greetings, help etc	Hi! I am your stock consultant. You can ask me about the stock information
	Supported companies	We support all companies in this page like Apple, Microsoft, etc.
Information query	what is the stock price of {company}	Currently in {company} , its stock price is {price} .
Prediction query	Should I buy {company} stock? can I buy the stock of	Currently in {company}, the stock is predicted to go up/down, we recommend you to buy it.

	Apple?	
Recommendation	Recommend companies	The top 3 recommended stocks are: {com1},{com2},{com3}

Appendix-4: Individual Project Report

individual project report-Huang Xinrun

1. personal contribution

I am the group leader of our project and responsible for determining our project scope and assigning sub-tasks to our members so that we produce our satisfied product finally. My contribution contains:

- The background research of project
- Overall system architecture designing
- Knowledge model design in stock movement prediction
- System user interface designing
- Integrating sub-modules to complete system with back end and front end
- System installation and user guide instruction

2. personal acquisition

After finishing our smart stock consultant project, the most useful knowledge for me is the data mining, optimization technique which is learned in Reasoning system module. Our project is the data driven task which summarize the rules learned from the tremendous history data and use learned knowledge to do reasoning on current data. Through this project, I have learned the fundamental process of building knowledge model. Define and quantify the business problem. We want to achieve higher investment return based on correctly predicting the stock market. Collect the historical data and chose the valuable indicators as features. Then design the knowledge model as knowledge representation to be stored and used for reasoning the prediction of stock. Another useful skill is the ability of integrating user friendly interface and our model. User can interact with our system through Web UI and automatic Q&A consultant. If the interface of system is not user friendly, people will not use it even the model is accurate and advanced.

3. practical usage

Therefore, in order to utilize the knowledge presentation and data mining techniques in our workplace. First, we should do some background research in project. We should define the project domain on business problem and what KPI we want achieve for the specific problem. Then we research possible solution or determine the problem is data driven task and then collect the historical data. After that we design our knowledge model to learn the rules from that collected data. Finally, the extracted knowledge model is our knowledge base which is used for reasoning.

As for the knowledge of optimization techniques, we need to define our goodness measurement based on the context of problem. Then we generate the possible solutions and find the best solution. And in order to interact with system through human language, we should process the human natural language. Analyzing the users' intents and provide the answer according to intents.

In general, I learn a lot of useful knowledge through our project and practical skills that will be used in our workplace.

individual project report-Wang Yangxi

1. **Personal contribution**

In our system, I have four contributions.

First, I made the market background analysis and thought about several ideas for our project, after discussing with my group members, we decided to develop system about stock consulting.

Second, after determining the idea, I started to think commercial value, then I did market analysis and thought about the solutions and advantages.

Third, during the coding period, I was mainly responsible for adding the Chatbot module to the system. I decided to use Dialogflow api to achieve this task. Plus, I wrote python code to cooperate with this api on the back-end. And I design the functions about cooperating with Chatbot module. In design, at first, I make some intents in the Dialogflow platform, then I send the intent which is analysed by the Dialogflow to the back-end. After that, I use the back-end design methods to answer different questions. Finally, I send the answer to the web page.

Fourth, I cooperate with my group members to achieve the writing of all reports, such as the final reports, proposal so on.

2. **Personal acquisition**

In this project, I think that the ability of finding the business problems and design the correspond artificial intelligent system for solving problem is most useful for me. Because we learn so much intelligent technologies, and we need to make value by

using them, which means that using intelligent technologies to deal with the actual problems.

Plus, another useful knowledge is the intelligent technology which is learned in the Intelligent Reasoning Systems module. Among these knowledge, we can find the best method to solve the actual problems.

3. **Practical usage**

I can use random forests for many other tasks, like classification, etc. For example, you can use to classify images which are token by cameras, to classify different people or objects, and make correspond labels on it.

I can add chatbot to any intelligent systems to make the systems more user-friendly, which can make systems easier to use in many situations.

In general, through working on this project, I learn lots of knowledge and practical skills which can be used in our workplace.

individual project report-Guo Fangjie

1. Personal contribution

In this project, I am responsible for the design of the back-end model and algorithm implementation of algorithm implementation of the smart stock advisor. My contribution contains:

- Data source selection and preprocessing
- Design and Implementation of Decision Tree and Random Forest Model
- Interface design of stock forecasting module
- Interface design of stock recommendation module
- Data set testing and evaluation

2. Personal acquisition

After completing our smart stock consultant project, the two most useful knowledge for me are the feature engineering and model optimization technique that are learned in the Intelligent Reasoning Systems module. Firstly, I learned the basic process of feature engineering. A well done feature engineering can maximize the effectiveness of the original data, which can often lead to a significant improvement in the effect and performance of the algorithm. Feature construction, feature extraction, and feature selection from raw data are all very important. In our project, after we obtained the original stock historical data, we need to perform data preprocessing and data cleaning, and select valuable indicators as features for training and prediction. Another useful

skill I have learned is model selection, training and tuning. Among the many machine learning models, we need to analyze and filter out the models that are suitable for our project, and adjust the parameters according to the understanding of the algorithm to make the model optimal.

3. Practical usage

Firstly, this project inspired me to the whole machine learning project process, which would help me understand the actual problem, and then abstracted as a mathematical problem that can be handled by machine learning.

As for the knowledge of feature engineering, like this project, we usually need to construct, extract, and select features from the original data. Preprocess data through normalization, discretization, factorization, missing value processing, collinearity removal, etc. It requires us to repeatedly understand the business to filter out salient features and discard non- salient features.

Moreover, after finish the work of feature engineering, we need to choose a suitable machine learning model for our reasoning process. We usually need to cross-validate and draw a learning curve for model diagnosis. After diagnosis, we need to further tune and iterate repeatedly.

In general, through the study and practice of this project, I am no stranger to the above-mentioned machine learning tasks, which are useful and practical knowledge for my future work.