School of Electrical, Electronic & Computer Engineering
Faculty of Engineering, Computing & Mathematics
The University of Western Australia

GENG5511 & GENG 5512

Engineering Research Project

Using Sentiment Analysis of tweets to predict stock movement

by

Qianwen Lu

22167601

Academic Supervisor: Professor Amitava Datta

Word Count: 6183

20th May 2019

Content

1. Introduction

1.1 Stock movement prediction

Stock price movement prediction has become an attractive economic research topic in engineering, finance, computer science, mathematics, and several other fields. However, the prediction of the stock price is full of uncertainty and is influenced by many factors. It is considered that stock price is a kind of finance time series [1]. Since the finance time series fluctuates with a lot of interference and it is a dynamic, selective, nonlinear, non-stationary change, it is difficult to forecast stock trend. Hence stock movement prediction is a challenging task.

The early stock movement prediction researchers ware based on the Efficient Market Hypothesis [2]. However, Timmermann [3] and Malkiel [4] found that it is hard to make predictions from the principle of the efficient market hypothesis due to the random work pattern of the stock price. [5] The rise and fall of stock price are affected by many factors such as political events, newspapers, quarterly and annual reports. In recent years, the researcher began to analyze those stock movement data and create models to predict stock movement. To improve the accuracy of stock movement prediction, scientists try to use sentiment analysis, data mining and machine learning techniques to create models. In the initial stage of the study of the stock market prediction, classical methods were used. But as the stock market is a non-stationary time series of data. It was not so effective. So non-linear machine learning techniques such as Artificial neural networks (ANN) and Support Vector Machine (SVM) are used widely. For example, Khedr, Salama, and Yaseen [6] created two efficient models using data mining technique and news sentiment analysis. The models predict stock market future trends with small error ratio and improve the accuracy of prediction.

1.2 Social media data and stock movement prediction

With the development of the internet, the methods of people get finance information have increased dramatically. In addition to news articles, analyst reports, and earnings statements, social media has been a significant way to gain access to business information. Social media, such as Twitter, Fackbook, Sina Weibo, contains various contents and spreads rapidly, which attracts an increasing number of users to share and comment on the posts. For instance, “Twitter reports on its homepage that it has 320 million monthly active users producing about 500 million tweets per day.” [7] Therefore, researchers begin their studies on finding the correlation between social media information and stock movements.

Social media analysis is a powerful method in stock movement prediction. Social media provides an informal platform for people to communicate and spread information, which breaks the barriers

of geography and society. [8] Compared with the features of the stock price mentioned above, data generated in social media have the same characteristics: dynamic, nonlinear, and unstructured. Therefore, based on the recent researches, the accuracy of stock price prediction can be increased by finding the correlation between existing social media data and stock price. Social media mining originates from the relevant field of data mining, which mines patterns from structured data instead of unstructured. It is also related to other fields like information retrieval, web mining, statistics, computational linguistics and natural language processing [9], [10]. For example, Sun, Lachanski and Fabozzi [11] use text mining technology to create a text analysis-based model for stock price prediction and found the model could increase the accuracy of stock prediction. Alsing and Bahceci [12] implemented a company-specific model by using machine learning and optimized Artificial Neural Network, which could predict stock price movement with 80% accuracy. Bollen, Mao and Zhang [13] analyzed the text content of daily Twitter and find that Twitter tweets are correlated to the value of the Dow Jones Industrial Average (DJIA) over time, with the accuracy of 86.7% in predicting the stock daily up and down changes in the closing values of the DJIA.

Therefore, in this paper, I tend to study the correlation between stock movements and related tweets. I choose the US stock market and collect stock and twitter dataset from different companies, which belong to nine different industries. I use sentiment analysis technology to analyze the sentiment of tweets and Artificial Neural Network technology to create a model to predict stock movement. I select seven stock related indicators and three sentiment related indicators as the input of the model. With the experiments, I find a strong correlation exists between the rise and falls in stock prices and the sentiments in tweets, with satisfied prediction accuracy. The rest of the paper is organized as follow. Section 2 introduces some related researches. Section 3 describes the methodology I used in this research. Section 4 is about my dataset. Section 5 is the implementation of the model. Section 6 assesses the results of the experiments. Finally, section 6 concludes my contribution and look forward to some future works.

2. Literature Review

In recent years, there have been lots of studies using social media data to predict stock movement. There are several aspects of technology have been used in related researches, such as sentiment analysis and machine learning. For social media data analysis, some previous researches focused on the articles of news and political reports. Nevertheless, with the development of Twitter, Facebook, Sina Weibo more studies are based on the posts and comments on these platforms. In addition, to create a suitable stock movement prediction model, various machine learning models have been used to set up the prediction model and the researches were more tended to improve the accuracy of the models.

2.1 Analysis of social media information

Sentiment Analysis refers to the automatic detection of emotional or opinionated statements in a text statement. Sentiment and perception are psychological constructs and thus difficult to measure in traditional methods. By analyzing the sentiment of social media information, researchers realized that social media data plays an important role in stock movement prediction. Text mining is used in sentiment analysis. Test mining refers to the statistical analysis of natural language data. It is applied to discover the sentiment of a textual article and classify text documents into sentiment categories (e.g. positive or negative sentiment categories). Positive emotion is likely to have a positive influence on stock movement and negative is true to negative.

Rechenthin, Street and Srinivasan [14] incorporated Yahoo Finance Message Board into the stock movement prediction. They tried to use various classification models to predict stock. They used the explicit sentiments and predicted sentiments obtained by a classification model with the bag-of-words and meta-features.

Nguyen and Shirai [15] built a model to predict stock price movement using the sentiment from social media. In the model, they added the sentiments of the specific topics of the company. Topics and related sentiments are automatically extracted from the texts in a message board. They compared the accuracy average over 18 stocks in a one-year transaction, achieving 2.07% better performance than the model using historical prices only. Furthermore, when comparing the methods only for the stocks that are difficult to predict, their method achieved 9.83% better accuracy than historical price method.

Sun, Lachanski and Fabozzi [11] use text mining technology to create a text analysis-based model for stock price prediction and found the model could increase the accuracy of stock prediction. The first important step for their model is the creation of a dictionary of terms through text mining. Their

dictionary was created by examining the top words for each year and combining them with the tickers of the 420 stocks. Then, they use a sparse matrix factorization (SMF) model for stock market prediction.

As twitter become famous, thousands of users post articles and comment news on it. Researches begin to analyze the correlation between Twitter tweets and stock market movements. Bollen, Mao and Zhang [13] obtained a collection of public daily tweets and used OpinionFinder as the sentiment analysis software package to determine sentence-level subjectivity. Then they used a self-organizing fuzzy neural network to predict the volatility of the Dow Jones Industrial Average (DJIA). By considering the sentimental information of related tweets, the prediction accuracy has increased by 13% and the total prediction model had an accuracy of 86.7% in predicting the stock daily up and down changes in the closing index of the DJIA.

Sprenger et al. [16] selected tweets related to the companies in the Standard & Poor's 100 index and labeled the tweets with the buy, hold or sell signals. They used a Naive Bayes classifier to extract the signals from the tweets automatically and calculated the bullishness through these signals. Finally, they found that a strategy based on bullishness signals could earn substantial abnormal returns.

Zhang et al. [17] measured collective hope and fear each day and analyzed the correlation between these indices and the stock market indicators. They used the mood words to tag each tweet as fear, worry, hope and soon. They concluded that the emotional tweet percentage significantly negatively correlated with Down Jones, NASDAQ, and S&P500, but had a significant positive correlation to VIX. However, they did not use their model to predict stock price values.

Mamaysky and Glasserman [18] show that text data can also be an indicator of market volatility. They aggregate over 360,000 articles on 50 large financial companies between 1996 and 2014 and examine sequences of n words, known as n-grams, classifying each as having positive sentiment or explicitly negative sentiment. They find that an increase in the unusual language of negative sentiment is subsequently followed by increased market volatility (measured by the VIX index) that lasts for several months at a time.

2.2 Machine learning technology in stock movement prediction

Machine learning is a method of data analysis that automates analytical model building. It is a branch of artificial intelligence based on the idea that systems can learn from data, identify patterns and make decisions with minimal human intervention. Stock Forecasting is one of the most common areas where machine learning is applied. Previous studies have indicated that machine models have positive impacts on forecasting stock market movement.

Liu and Zhou [19] adopted a new algorithm which integrates K-Nearest Neighbor (KNN) and Artificial Neural Network (ANN). This new algorithm can predict stock price more accurately by studying the time series. Firstly, they use KNN to come up with a weighted coefficient. Then, use a BP neural network to optimize the weighted coefficient. They used their KNN-ANN model to predict the 6 days' stock price of Central China Science and Technology. The results show that the predicting model based on KNN-ANN algorithm can do better in estimating the stock price.

Kara, Boyacioglu, Baykan [20] predicted the direction of stock price index movement using Artificial Neural Networks and Support Vector Machines. They used the Istanbul Stock Exchange to predict the trend of the stock price. Using ANN they have got 75.74 % accuracy and using polynomial SVM 71.52 %.

Şenol and Özturan [21] studied the stock market index in Turkey by applying seven different machine learning models for predicting the direction. They concluded that ANN could be one of the most useful techniques for forecasting.

According to the previous researches, ANN model can be popularly claimed to be a useful technique for stock movement prediction due to its ability to capture functional relationships among the original data even though the underlying relationships are unknown or hard to describe [22]. Application of ANN has become the most popular machine learning method and it has been proven that such an approach performs much well than conventional methods.

Based on the previous researches mentioned above, I attempt to select different companies from the US stock market and analyze sentiment of those companies' tweets. Ultimately, I apply stock related information and sentiment analysis results as the input of an ANN model to forecast the direction of the stock market index.

3. Methodology

3.1 Sentiment analysis tool: TextBlob

In this research, I apply TextBlob [23], which is a Python library for processing textual data, to analyze the sentiment of tweets. TextBlob provides a simple API for diving into common natural language processing (NLP) tasks such as part-of-speech tagging, noun phrase extraction, sentiment analysis, classification, translation, and more. TextBlob stands on the giant shoulders of Natural Language Toolkit (NLTK) and pattern and plays nicely with both. It has several different features: Noun phrase extraction, Part-of-speech tagging, Sentiment analysis, Classification (Naive Bayes, Decision Tree), Language translation and detection powered by Google Translate, Tokenization (splitting text into words and sentences), Word and phrase frequencies, Parsing, n-grams, Word inflection (pluralization and singularization) and lemmatization, Spelling correction, Add new models or languages through extensions, WordNet integration. Sentiment analysis is one of its features. The *textblob.sentiments* module contains two sentiment analysis implementations, PatternAnalyzer (based on the pattern library) and NaiveBayesAnalyzer (an NLTK classifier trained on a movie reviews corpus). The default implementation is PatternAnalyzer, but developers can override the analyzer by passing another implementation into a TextBlob's constructor. For instance, shown in Figure 3.1.1, the NaiveBayesAnalyzer returns its result as a namedtuple of the form: Sentiment (classification, p_pos, p_neg). It will return the sentiment "score" of the input text, with the range of 0 to 1, 0 means completely negative, and 1 means completely positive.

```
>>> from textblob import TextBlob
>>> from textblob.sentiments import NaiveBayesAnalyzer
>>> blob = TextBlob("I love this library", analyzer=NaiveBayesAnalyzer())
>>> blob.sentiment
Sentiment(classification='pos', p_pos=0.7996209910191279, p_neg=0.2003790089808724)
```

Figure 3.1.1 An instance using textblob.sentiments by implementing NaiveBayesAnalyzer

3.2 Artificial Neural Network

Artificial neural networks (ANN) is a computing system inspired by biological neural networks. An ANN is based on a collection of connected units or nodes called artificial neurons, which is like the neurons in a biological brain. In each connection, like the synapses in a biological brain, can transmit a signal from one artificial neuron to another. [24] An artificial neuron that receives a signal can process it and then signal additional artificial neurons connected to it. In basic ANN model implementations, the signal at a connection between artificial neurons is a real number, and the output of each artificial neuron is computed by some non-linear function of the sum of its inputs. The connections between artificial neurons are called 'edges'. Artificial neurons and edges typically

have a weight that adjusts as learning proceeds. The weight increases or decreases the strength of the signal at a connection. Artificial neurons may have a threshold such that the signal is only sent if the aggregate signal crosses that threshold. Propagation function is used in computing the input value to the neuron from the outputs of predecessor neurons and usually has a bias value. The function generally has the form:

$$p_j = \sum_i o_i(t)\, w_{ij} + w_{0j} \quad (1)$$

Where p_j is the input neuron, o_i is the output neuron, w_{ij} is the connection represents as weight and w_{0j} is the bias. Typically, artificial neurons are aggregated into layers. Different layers may perform different kinds of transformations on their inputs. Signals travel from the first layer (the input layer) to the last layer (the output layer), possibly after traversing the layers (the hidden layer) multiple times. Figure 3.2.1 shows a simple ANN model.

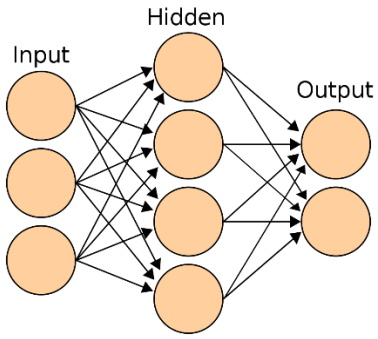

Figure 3.2.1 A simple artificial neural network model

As a machine learning model, ANN is capable of learning and they need to be trained. The learning strategy is a rule or an algorithm which modifies the parameters of the neural network, in order for a given input to the network to produce a favored output. This learning process typically amounts to modifying the weights and thresholds of the variables within the network [25]. There are several learning strategies: supervised learning, unsupervised Learning, reinforcement learning. Supervised learning involves a teacher that is a scholar than the ANN itself. For example, the teacher feeds some example data about which the teacher already knows the answers. It can be used in pattern recognizing. The ANN comes up with guesses while recognizing. Then the teacher provides the ANN with the answers. The network then compares it guesses with the teacher's "correct" answers and makes adjustments according to errors. Unsupervised Learning is required when there is no example data set with known answers. It is used for searching for a hidden pattern. In this case, clustering dividing a set of elements into groups according to some unknown pattern is carried out based on the existing data sets present. Reinforcement Learning built on observation. The ANN makes a decision by observing its environment. If the observation is negative, the network adjusts

its weights to be able to make a different required decision the next time.

Since artificial neural networks have high performance on training complexity input data and could approximate any unknown function to any degree of the desired accuracy, the ANN model is used in this study. The model consists of the input layer, the hidden layer, and the output layer, each of which is connected to the other in the same sequence. The input layer corresponds to the stock related variables and related sentiment data. The hidden layer is used for capturing the nonlinear relationships among variables. Each neuron in the hidden layer has the inputs multiplied by a weight, and all inputs are summed. The value is passed by using an activation function. In this study, the output layer consists of only one neuron that represents the predicted movement of the daily stock index.

4. Dataset

4.1 Stock data collection

In this study, I attempt to predict the US stocks. The research data used in this study is the historical stock prices of nine companies from nine different industries. The total number of trading days is 565 days, from January 2014 to March 2016. All these historical stock trading prices are extracted from Yahoo Finance. The list of company names and stock symbols is shown in Table 4.1.1. For each transaction date, there are open, high, low, close, adjusted close prices and transaction volume. I divide the stock data into two parts, 75% of the data is used for the training dataset and 25% is considered as the testing dataset.

Industry	Stock symbol	Company
Basic Materials	$XOM	Exxon Mobil Corporation
Consumer Goods	$AAPL	Apple Inc.
Healthcare	$JNJ	Johnson & Johnson
Services	$AMZN	Amazon.com, Inc.
Utilities	$EXC	Exelon Corporation
Conglomerates	$HRG	HRG Group, Inc.
Financial	$BAC	Bank of America Corporation
Industrial Goods	$GE	General Electric Company
Technology	$MSFT	Microsoft Corporation

Table 4.1.1 The list of company names and stock symbols

4.2 Twitter tweets data collection

Generally, researches get tweets using Twitter API, which is provided by Twitter officially. However, there are lots of limitations such as download rate limit, tweets quantitative limit, GET and POST request limit. As a result, I get tweets data from Kaggle, which is an open source website supporting a variety of dataset publication formats. The dataset is a total of 400,000 tweets over a period of January 1st, 2014 to March 31st, 2016. I need to get company-related tweets from the whole data set. The tweets were collected keywords of a company. For example, for apple company, the keywords are like $AAPL, #Apple Inc., #iPhone, #iMac, etc. Not only the altitude of the public about the company's stock but also the opinions about products and services offered by the company would have a significant impact and are worth to study. The news on Twitter about the company and tweets regarding the product releases were also included. Based on the principles, these company-related tweets could represent the exact emotions of the public about the company over a period of time. Table 2 shows the number of collected tweets for companies.

Stock symbol	Company	Number of Tweets
$XOM	Exxon Mobil Corporation	3904
$AAPL	Apple Inc.	20788
$JNJ	Johnson & Johnson	2343
$AMZN	Amazon.com, Inc.	7212
$EXC	Exelon Corporation	1284
$HRG	HRG Group, Inc.	937
$BAC	Bank of America Corporation	3570
$GE	General Electric Company	3943
$MSFT	Microsoft Corporation	8603

Table 4.2.1 The number of collected tweets for the companies

4.3 Data processing

It is well known that the stock market closes on weekends and public holidays, so the collected stock prices data miss the stock prices of closed dates. Goel [26] has put forward a simple technique to fix the missing data. There is a function called concave, which is usually used in the stock data processing. So, if there are some missing stock prices value between two days, the stock value on the first day is x and the next value present is y. The first missing value is approximated to be $\frac{y+x}{2}$ and as well as the following all the gaps.

After the collection of useful tweets, each company's tweets data has a separate file, which is easy to process. In addition, the original tweets dataset is saved in JSON format. Therefore, I convert the JSON format to CSV format, which is the same as the stock dataset, by using simple python scripts. Furthermore, for each collected tweet, there are many attributes which are not necessary for this research. The useful columns are dates of tweets and texts of tweets. In addition, the texts of tweets consist of many acronyms, emoticons and unnecessary data like URLs. So, the tweets need to process to well represent the correct emotions of the public. Python provides a strong library, Pandas, to process data structures and analyze data. Firstly, I collect the two useful columns: dates and texts. For the dates, the date represents “Wed Jan 01 03:59:03 +0000 2014”. It should be normalized to the same style as the date column of the stock dataset as “2014/1/1”. Secondly, for texts processing, I employed two stages of filtering: removing noisy tweets and deleting special characters by using regex matching. Noisy tweets refer to the texts just have one or two words which could not express any emotion. For example, there is a text “#Apple #Microsoft”. This is a noisy tweet and needs to remove from the tweets list. Then, deleting special characters, such as URLs, the most common method is that using the regex matching to find out the URLs and delete them. In addition, the texts often consist of hashtags (#) and @ addressing other users. They also need to be replaced by suitably. For example, #AMAZON is replaced with AMAZON and @Jason is replaced with USER. Also,

there is a kind of prolonged word showing intense emotions like cooooooool, which is replaced with cool. After these stages, the tweets dataset is ready for sentiment analysis.

5. Sentiment analysis and Modelling

5.1 Sentiment analysis

I apply TextBlob as the sentiment analyzers to analyze the sentiment of the collected tweets data. The purpose of sentiment analysis is to get a sentiment score for each tweet and count the number of positive tweets each day. There is a model in TextBlob called textblob.sentiments, which is used to output the scores for each tweet. It will return the "score" of the input text, with the range of 0 to 1, 0 means completely negative, and 1 means completely positive. For example, the input twitter text is "Microsoft acquires Parature to add leading customer self-service suite to Microsoft Dynamics CRM", with the output of positive score 0.86879467 and negative score 0.13120533 (1 – positive score). The positive score and the negative score are the added two new columns in the sentiment dataset CSV files. There is a python script applied to read the tweets line by line and analyze the sentiment automatically. After obtaining all the sentiment score of the tweets, the normalization step is important. There are three features added to the sentiment dataset: average positive score per day, number of positive tweets per day and number of negative tweets per day. The average positive score for each day is calculated as the following formula:

$$\text{avg pos score} = \frac{1}{n}\sum_{i=1}^{n} P_i \ (i = 1, 2, 3, \cdots, n) \qquad (2)$$

where P_i is the positive score of each tweet on the same day, n is the number of tweets each day. However, due to the limitation of Twitter tweets dataset, there are some dates without any tweet for several companies. I just set the average positive score in this day as 0.5 and both 0 for the number of positive and negative tweets. Fortunately, these dates are not so many, which is less than 15 day in a total of nine companies. In addition, I count the number of positive and negative tweets per day and save the numbers as two new indicators of the tweets' sentiment.

As a result, the final sentiment dataset has seven attributes: dates, texts, positive score, negative score, avg pos score, num of pos and num of neg. The next step is to merge stock dataset and sentiment dataset. I just copy the last three features (avg pos score, num of pos and num of neg) in sentiment dataset and paste them to the stock dataset, since the dates are a one-to-one correspondent. Figure 5.1.1 is a row of the combined dataset. This dataset is used for stock movement modeling.

A	B	C	D	E	F	G	H	I	J
Date	Open	High	Low	Close	Volume	Adj Close	Sentiment	Num of Pos	Num of Neg
2016/4/18	36.240002	37	35.880001	36.52	17723000	36.52	0.48298993	10	2
2016/4/15	37.130001	37.150002	36.419998	36.509998	19016200	36.509998	0.71717062	20	7

Figure 5.1.1 A simple row of combined dataset

5.2 Modelling

According to the previous researches, the artificial neural network performs a significant advantage in modelling stock movement prediction. In this study, I apply the backpropagation algorithm when creating the ANN model. The artificial neural network is created based on keras, which is a python deep learning library.

5.2.1 Backpropagation algorithm

The backpropagation algorithm is a widely applied classical learning algorithm for neural networks [27,28]. Backpropagation, another way to say "backward propagation of errors," is a calculation for supervised learning of artificial neural networks utilizing gradient descent. Given an artificial neural network and an error function, the strategy ascertains the gradient of the error function regarding the neural network's weights. It is a speculation of the delta manages for perceptrons to multilayer feedforward neural networks. In this research, backpropagation uses Gradient Descent Optimizer for reducing error.

The "backward" some portion of the name originates from the way that computation of the gradient continues backward through the network, with the gradient of the last layer of weights being ascertained first and the gradient of the principal layer of weights being figured last. Fractional calculations of the gradient from one layer are reused in the calculation of the gradient for the past layer. This backward stream of the mistake data takes into account the productive calculation of the gradient at each layer versus the innocent approach of computing the gradient of each layer independently.

As shown in Figure 5.2.1, the BP process determines the weights for the connections among the nodes and their biases based on the input training data. For instance, w11 indicates the weight between the first node of the input layer and the first node of the hidden layer. θ1 represents the bias of the first node in the hidden layer. Initially, the value weights and biases are assigned in constant values. When training data, the error between the predicted and actual output values is back propagated via the network for updating the weights and biases repeatedly. When the error is less than a specified value or when the termination criterion is satisfied, training is considered to be completed and the weights and bias values of the network are stored.

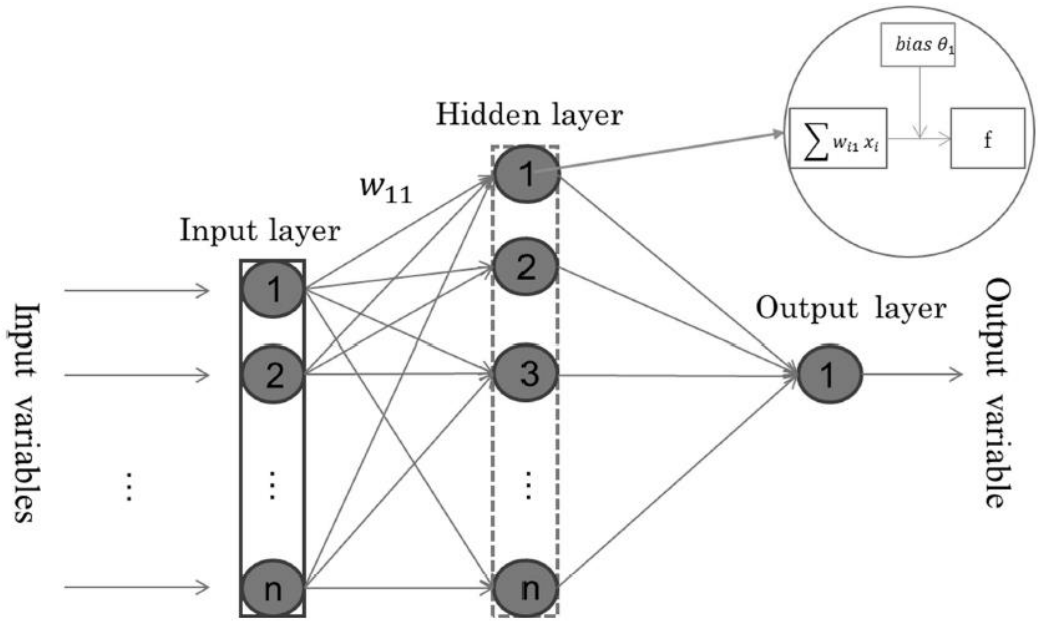

Figure 5.2.1. The architecture of the back propagation neural network.

5.2.2 Layers in ANN model

There are three layers in this study: input layer with ten input variables, a hidden layer with activation functions to calculate weights and biases, output layer with the predicted of the daily stock movement. I will explain the algorithm and layers in detail in the following sections.

In the light of previous studies, the researchers always use various stock-related technical indicators as input variables in the construction of prediction models to forecast the direction of movement of the stock price index [29]. Most financial managers and investors agree on these efficient technical indicators and exploit them as a signal for forecasting future trends. Technical indicators of the input variables are usually used to predict future trends, and they are derived from the real stock composite index. Based on the prior studies [22, 30, 31], I select seven stock related technical indicators as the input variables as shown in Table 5.2.2. This table lists the selected features and their formulas.

Name of indicators	Formulas
Weighted n (10 here) day moving Average	$\frac{(10)C_t + (9)C_{t-1} + \cdots + C_{t-9}}{n + (n-1) + \cdots + 1}$
Momentum	$C_t - C_{t-9}$
Stochastic %K	$\frac{C_t - LL_{t-(m-1)}}{HH_{t-(n-1)} - LL_{t-(m-1)}} \times 100$
Stochastic %D	$\frac{\sum_{i=0}^{n-1} K_{t-i}}{10}\%$
Relative Strength Index (RSI)	$100 - \frac{100}{1 + (\sum_{i=0}^{n-1} \frac{UP_{t-i}}{n})/(\sum_{i=0}^{n-1} \frac{DW_{t-i}}{n})}$

Moving Average Convergence Divergence (MACD)	$MACD(n)_{t-1} + \frac{2}{n+1} \times (DIFF_t - MACD(n)_{t-1})$
Accumulation/Distribution (A/D) Oscillator	$\frac{H_t - C_{t-1}}{H_t - L_t}$

Table 5.2.2 The stock related features and their formulas

C_t is the closing price, L_t is the low price and H_t is the high price at time t. $DIFF_t = EMA(12)_t - EMA(26)_t$, EMA is exponential moving average, $EMA(k)_t = EMA(k)_{t-1} + \alpha \times (C_t - EMA(k)_{t-1})$, α is a smoothing factor which equal to $\frac{2}{k-1}$, k is the time period of *k-day* exponential moving average, HH_t and LL_t implies highest high and lowest low in the last t days, respectively. $M_t = \frac{H_t + L_t + C_t}{3}$, $SM_t = \frac{\sum_{i=1}^{n} K_{t-i+1}}{n}$, $D_t = \frac{(\sum_{i=1}^{n} |K_{t-i+1} - SM_t|)}{n}$, UP_t means upward price change while DW_t is the downward price change at time t.

The purpose of this research is analyzing the correlation between the sentiment of tweets and stock movement, so the input variables should also include sentiment related indicators. As mentioned in the dataset section, I calculate three features of tweets for each day: average positive score, number of positive tweets and number of negative tweets. Therefore, the input variables contain seven stock related indicators and three sentiment related indicators.

In this artificial neural network, for each neuron, it has a label and receives input from predecessor neurons. The label is the trend for stock movement for the next day, 1 for up and -1 for down. In addition, the network consists of connections, each connection transferring the output of a neuron to the input of a new neuron and each connection is assigned a weight. An activation function computes the new activation at a given time t+1 from the previous activation, an output function computes the output from the activation. Sometimes a bias term is added to the total weighted sum of inputs to serve as a threshold to shift the activation function. In this study, the input layer and hidden layer use ReLu activation function and the output layer uses sigmoid activation function. Relu activation function refers to the rectified linear unit and it represents the following argument:

$$\mathrm{f(x)} = \begin{cases} 0 & for\ x < 0 \\ x & for\ x \geq 0 \end{cases} \qquad (3)$$

The main reason behind using ReLu function is that it creates more sparse data when W*X + B < 0, and other reason is it reduces chances of the gradient to vanish. The output layer has only one neuron and it gives almost the same accuracy, so I have taken sigmoid function to obtain a nonlinear model. Also, I have normalized the dataset. Sigmoid function often refers to the special case of the logistic function shown in the first figure and defined by the formula:

$$f(x) = \frac{1}{1 + e^{-x}} = \frac{e^{x}}{1 + e^{x}} \quad (4)$$

The prediction performance Hit ratio and prediction result are evaluated using the following equation:

$$\text{Hit ratio} = \frac{1}{n} \sum_{i=1}^{n} P_i \ (i = 1, 2, 3, \cdots, n) \quad (5)$$

$$P_i = \begin{cases} 1, (y_{t+1} - y_t)(\hat{y}_{t+1} - \hat{y}_t) > 0 \\ 0, \ otherwise. \end{cases} \quad (6)$$

where P_i is the prediction result for the ith trading day, which is defined by equation 5. The variable y_t represents the actual value of the closing stock index for the ith trading day, and $\hat{y}_t$ is the predicted value for the ith trading day. The variable n indicates the number of test samples. Therefore, the ANN model has been done.

6. Results and discussion

6.1 Experiments and results

Firstly, I will indicate some initial settings of the parameters in the BP-ANN model and it is shown in Table 6.1.1.

Parameters	Value	Definition
n	10	Number of neurons in the hidden layer of the ANN model
ep	8000	Number of iterations for the ANN model
mc	0.4	Momentum constant of the Ann model
I	0.1	Value of learning rate of the ANN model

Table 6.1.1 Description of parameters used in the hybrid model.

One of the purposes of this research is to check whether there is any correlation between the sentiment of tweets and stock movement prediction. To verify the topic of this study is true, I firstly use only seven stock related variables as the input variables to training the model. The prediction accuracy is shown below. From there Table 6.1.2, it is clear that the total training accuracy of the nine company is around 71.5% and testing accuracy 67.34%, which means the prediction results of using stock related indicators as the input variables for different companies are nearly the same. There are no obvious differences in different companies' stock movement prediction.

Industry	Stock symbol	Training accuracy	Testing accuracy
Basic Materials	$XOM	71.23%	67.88%
Consumer Goods	$AAPL	71.56%	68.97%
Healthcare	$JNJ	71.86%	67.23%
Services	$AMZN	70.69%	68.15%
Utilities	$EXC	72.35%	69.14%
Conglomerates	$HRG	70.72%	66.17%
Financial	$BAC	71.89%	65.28%
Industrial Goods	$GE	72.01%	69.21%
Technology	$MSFT	71.58%	67.92%

Table 6.1.2 The stock movement prediction result based on stock related input variables only

Then, I change the input variables to ten, which contains all the stock related indicators and sentiment related indicators. I have explained all the indicators in the previous section. Table 6.1.3 shows the prediction accuracy by using these ten input variables.

Industry	Symbol	Number of tweets	Training accuracy	Testing accuracy
Basic Materials	$XOM	3904	80.28%	75.26%
Consumer Goods	$AAPL	20788	**85.30%**	**80.13%**
Healthcare	$JNJ	2343	77.07%	72.14%

Services	$AMZN	7212	81.36%	78.03%
Utilities	$EXC	1284	75.77%	67.76%
Conglomerates	$HRG	937	**72.72%**	**67.37%**
Financial	$BAC	3570	78.58%	74.86%
Industrial Goods	$GE	3943	80.45%	76.51%
Technology	$MSFT	8603	83.29%	79.82%

Table 6.1.3 The stock movement prediction result based on all the input variables

Compared with Table 6.1.3, the prediction accuracy of companies has a significant difference. The total prediction accuracy goes up 2% to 10%. The Apple company, which has the largest tweets dataset, has the highest stock movement prediction training and testing accuracy of 85.30% and 82.13%. For apple company, the total prediction accuracy has increased nearly by 10%, especially the training accuracy increasing by 13%. On the other hand, HRG company has the lowest prediction accuracy, which contains the smallest tweets dataset, the increase ratio is the smallest. Also, for other companies, the accuracy of their stock movement predictions was proportional to the number of tweets they had.

6.2 Discussion

Based on Table 6.1.2 and Table 6.1.3, the prediction accuracy has an obvious increase when using combined stock related indicators and sentiment related indicators, which means by analyzing the sentiment of company related tweets and use them in stock prediction, the accuracy of stock movement prediction could increase. There is a strong correlation between the sentiment analysis of tweets and stock movement. In addition, it is clearly shown in the results that the greater number of tweets the higher prediction accuracy. With large tweets dataset, the prediction accuracy increases rapidly.

On the other hand, in light of the previous studies, many researchers have compared ANN with SVM. For example, Kim [32] applied SVM to predict the stock price index, and compare it with the backpropagation neural networks. Their study shows that SVM outperforms BP neural networks in stock movement forecasting. We suppose that researchers usually focus on the parameter selection of BP neural networks when they compare it with other models. If they combine the selection of input variables and the optimal adjustment of the weights and biases of the ANN model, the optimized ANN model may still provide a promising alternative to stock market prediction.

7. Conclusion and Future work

In conclusion, for this research, I applied stock related indicators and sentiment related indicators to predict the movement of the next day's stock index for nine companies. Firstly, I collected stock data of the nine company from Yahoo finance and original Twitter tweets from Kaggle. After processing the data, I analyze the sentiment of tweets by TextBlob and obtained three sentiment related indicators. I adjusted the weights and biases of the ANN model using the backpropagation algorithm. By studying previous researches, I decided to use seven stock related indicators. I combined the two indicators together as the input variables of the BP-ANN model. I implemented two experiments, with the different inputs. One only used seven stock-related input and the other one uses all the stock related and sentiment related indicators. The result of my prediction model has a satisfying accuracy in prediction. Using sentiment analysis of tweets has a positive effect on the accuracy of stock movement prediction improvement. In addition, with larger tweets dataset, the prediction accuracy will be higher.

For further study, the prediction performance could be improved, and optimized ANN model could be applied in stock movement prediction. There are three ways to extend the research. The first method is to combine more stock related indicators and sentiment related indicators. For the sentiment indicators, some textual input can be tested. Second, optimal methods, such as genetic algorithm, KNN and RNN models, may also have a positive effect of improving the accuracy of prediction. Lastly, a larger dataset is necessary. In fact, the number of related tweets for the nine company should be very great. With larger dataset, the prediction accuracy could be improved.

Reference

[1] R. S. Tsay. Analysis of financial time series [M]. *John Wiley & Sons*, 2005.

[2] E. F. Fama. The behavior of stock-market prices[J]. *The journal of Bussiness*, 1965, 38(1):34-105.

[3] A. Timmermann, C. W. Granger. Efficient market hypothesis and forecasting [J]. *International Journal of Forecasting*, 2004, 20(1): 15-27.

[4] B. G. Malkiel B G. The efficient market hypothesis and its critics [J]. *Journal of economic perspectives*, 2003, 59-82.

[5] M. J. Wang, M. Q. Wang. Using Social Media Mining Technology to Assist in Price Prediction of Stock Market. *2016 IEEE International Conference on Big Data Analysis*, 2016, pp.1-4

[6] Mahanta, T.N. Pandey, A.K. Jagadev, S. Dehuri, Optimized radial basisfunctional neural network for stock index prediction, *2016 InternationalConference on Electrical, Electronics, and Optimization Techniques (ICEEOT)*, 2016, 1252–1257.

[7] E. F. Fama. The behavior of stock-market prices[J]. The journal of Bussiness, 1965, 38(1):34-105.

[8] M. Thelwall, K. Buckley and G. Paltoglou, Sentiment strength detection for the social web, *Journal of the American Society for Information Science and Technology*, 63(1) (2012), 163–173.

[9] TANG L, LIU H. Community detection and mining in social media [J]. *Synthesis Lectures on Data Mining and Knowledge Discovery*, 2010, 2(1): 1-137.

[10] CORLEY C D, COOK D J, MIKLER A R, et al. Text and structural data mining of influenza mentions in web and social media [J]. *International journal of environmental research and public health*, 2010, 7(2): 596-615.

[11] A. Sun, M. Lachanski, F.J. Fabozzi. Trade the tweet: Social media text mining and sparse matrix factorization for stock market prediction. *International Review of Financial Analysis*, 2016, Vol.48, pp.272-281

[12] O. Alsing, O. Bahcecl. Stock Market Prediction using Social Media Analysis. 2015

[13] J. Bollen, H. Mao, X. Zeng, Twitter mood predicts the stock market. *Comput.Sci*. 2 (1) (2011) 1–8.

[14] Rechenthin, M., Street, W.N., & Srinivasan, P. Stock chatter: using stock sentiment to predict price direction. *Algorithmic Finance*, 2013, 2(3), 169–196.

[15] T. Nguyen, K. Shirai, Topic modeling based sentiment analysis on social media for stock market prediction. *Proceedings of the 53rd Annural Meeting of the Association for Computational Linguistics*, 2015.

[16] T.O. Sprenger, A. Tumasjan, P.G. Sandner, I.M. Welpe, Tweets and trades: the information

content of stock microblogs. *European Financial Management*, 20(5), 926–957 (2010)

[17] X. Zhang, H. Fuehres, P.A. Gloor, Predicting stock market indicators through twitter "I hope it is not as bad as I fear". *Social and Behavioral Sciences* 26(26), 55–62 (2011)

[18] H. Mamaysky & P. Glasserman, Does unusual news forecast market stress? *Working papers in financial research*. (2015)

[19] T. Wilson, J. Wiebe, P. Hofimann, Recognizing contextual polarity in phrase-level sentiment analysis, *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, 2005; 347–354,

[20] Y. Kara, M.A. Boyacioglu, K. Baykan. Predicting direction of stock price index movement using artificial neural networks and support vector machines: The sample of the Istanbul Stock Exchange. *Expert Syst Appl*. 2011; 38(5):11–19.

[21] D. Şenol D, M. Özturan. Stock price direction prediction using artificial neural network approach: the case of Turkey. *Artif Intell*. 2008; 1(2):70–77.

[22] Qiu, Mingyue, et al. Predicting the Direction of Stock Market Index Movement Using an Optimized Artificial Neural Network Model. *PLOS ONE*. 2016; 5(11)

[23] TextBlob official website https://textblob.readthedocs.io/en/dev/

[24] Butterfield, Andrew, and Ngondi, Gerard Ekembe. "Artificial Neural Network." *A Dictionary of Computer Science*. 2016. Web.

[25] Ojha Varun Kumar et al. "Metaheuristic design of feedforward neural networks: A review of two decades of research". *Engineering Applications of Artificial Intelligence*. 2017; 60: 97–116

[26] Mittal, Anshul, and Arpit Goel. "Stock prediction using twitter sentiment analysis." *Standford University CS229*. 2012

[27] R.S. Sexton, J.N. Gupta. Comparative evaluation of genetic algorithm and backpropagation for training neural networks. *Inf Sci*. 2000; 129(1):45–59.

[28] P.J. Werbos. The roots of backpropagation: from ordered derivatives to neural networks and political forecasting. 1st ed. John Wiley & Sons; 1994.

[29] C. Huang, C. Tsai. A hybrid SOFM-SVR with a filter-based feature selection for stock market forecasting. *Expert Syst Appl.* 2009; 36(2):29–39.

[30] E.W. Saad, D.V. Prokhorov, D.C. Wunsch. Comparative study of stock trend prediction using time delay, recurrent and probabilistic neural networks. *IEEE Trans Neural Netw.* 1998; 9(6):56–70.

[31] G. Armano, M. Marchesi, A. Murru. A hybrid genetic-neural architecture for stock indexes forecasting. *Inf Sci.* 2005; 170(1):3–33.

[32] T. Jo. VTG schemes for using back propagation for multivariate time series prediction. *Appl Soft Comput.* 2013; 13(5):692–702.