Stock Market Prediction based on Time Series Data and Market Sentiment

Tina Ding
Northwestern University
Apt 301, 1940 Sherman Avenue
Evanston, IL 60201
1-847-702-4609
xiaotianding1.2013@u.northwestern.edu

Vanessa Fang
Northwestern University
2133 1/2 ridge Ave, #2D
Evanston, IL 60201
1-703-405-0688
vanessafang2014@u.northwestern.edu

Daniel Zuo
Northwestern University
626 University Place
Evanston, IL 60201
1-847-220-3178
pengzuo2014@u.northwestern.edu

ABSTRACT

In this project, we would like to create a system that predicts stock market movements on a given day, based on time series data and market sentiment analysis. We will use Twitter data on that day to predict the market sentiment and S&P 500 values to perform analysis on historical data.

1. INTRODUCTION

Stock market predication has always been an interesting topic among researchers. We found the idea of combining market data with public sentiment to predict market movement particularly interesting when addressing this topic. We believe that such a combination could help more accurately predict stock market movement. We seek to find the most relevant historical data attributes, the best learning method, and whether the addition of a public sentiment attribute is helpful in the prediction of stock market movement.

2. OVERVIEW

2.1 Objective

Given the time series data and Twitter data from January 2008 to April 2010, we will construct a system to predict stock market movement (up, same, down) on a given day, based on the key attributes computed from the historical data from the past few days and market sentiment. Given these inputs, the model will be expected to output a prediction of S&P index movement for a given day.

We would like to train the system on three models and compare the performances of these three models. Furthermore, we would like to determine whether the addition of a public sentiment attribute is beneficial in the prediction of market movement. To help us examine these problems, we would like to raise three questions:

1. Which classification method is the most accurate in predicting market movement on a given day? SVM, Logistic Regression, or Neural Networks?
2. Is the use of 1, 5, 10, or 30 days of prior market day most helpful in the prediction of market movement?
3. Does the addition of public sentiment, in this case from Twitter, help in the prediction of market movement?

2.2 General Approach

We will train the system on three models, SVM, logistic regression, and neural networks. Afterwards, we will compare the performances of these three learning models, with and without market sentiment, and determine which is the most effective. We will use N-fold cross validation on data in the timeframe of January 2008 and April 2010 to measure the performance of the system.

2.3 Method and Software Usage

We decided to use the Python NLTK (Natural Language Toolkit) for our sentiment analysis [7]. The NLTK is an open source suite that provides some useful tools and libraries for text processing. The NLTK package also includes a number of trainable classifiers, including a Naive Bayes classifier with built-in training and classifying methods.

2.4 Dataset

For time series data analysis, we directly imported the prices for S&P 500 from January 2008 to April 2010 from Yahoo! Finance into Excel spreadsheet.

For sentiment analysis, we obtained the *Twitter Census: Stock Tweets dataset* from Infochimps, a privately held company that offers a "data marketplace" that gives users access to public and proprietary data sets [8]. This dataset includes 2.3 million stock tweets. The data set provides the timestamp, ticker symbol, tweet ID, and keywords for each tweet. An example is provided below:

20090323173524 $TAZ 1376714687 make,money,buy

After inspecting the dataset, a few concerns were raised. 1) The tweets before 2008 were sparse and did not contain representative keywords; 2) Many of the keywords extracted are irrelevant for our purposes; 3) The data set does not provide original text of tweets, instead it features only extracted keywords, resulting in a loss of context.

We decided to modify the raw dataset in order to address some of these concerns. We first removed all tweets before 2008. We then created a bag of 80 relevant words for market movement prediction. Using this bag, we went through the data set and for each tweet, extracted the timestamp and relevant keywords from our bag.

The third concern mentioned was not something we addressed in this project, however, it is an important one to take into consideration. It is possible that the lack of context for our examples prevents us from accurately determining the sentiment from each tweet and each day.

To train the Naïve Bayes Classifier in NLTK, we manually prepared a training set that contains 295 labeled tweets. An example instance is provided:

$SINA in short now, seems not much interest. -1

3. ANALYSIS & RESULTS

3.1 Times Series Data Analysis

Based on S&P movement chart, we found that the magnitude of index movement varies a considerable amount and more than a quarter of the movements are within [-5,5]. Therefore, we believed labeling movements into 3 categories would provide more useful information on market movement. Since the magnitude of movement between [-5, 5] is relatively small, we chose to label this as "same" or "not moving." We finally have 3 labels for S&P index movement: Up, Down, and Same (relative to previous day).

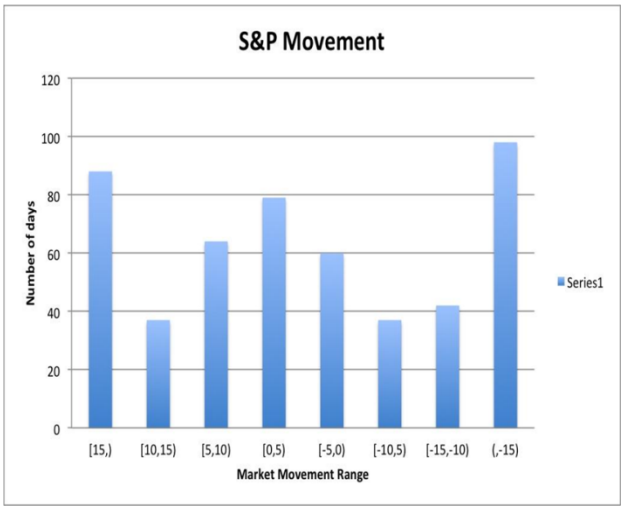

Figure 1. Market Movement Histogram

Label	Up	Same	Down	Total
Count (Percentage)	186 (37.34%)	137 (27.51%)	175 (35.14%)	498 (100%)

Table 1. Data distribution

We used the following attributes to predict S&P movement. We chose our attributes based on "Prediction of Closing Stock Prices"[6]. For each attribute, there is a threshold or criteria that indicate the S&P will go up or down in the next day. Since we have 3 classes of index movements, we decided to change the number of labels of each attribute from 2 to 3.

Based on the original threshold value and criteria, we divided each attributes into three classes: we label it as "-1" if it strongly indicates the market will go down, "1" if it strongly indicates the market will go up, and "0" if it does not have strong indication. We chose the threshold values so that each attribute are approximately evenly divided.

We also wanted to compare the performance of using data of different time period. Therefore, we adjusted the calculations of attributes so that their values reflect the information of a certain time period. Here are the calculation and labeling of each attribute for t-day period data:

Attribute 1: On Balance Volume-Movement

On Balance Volume (OBV) measures buying and selling pressure as a cumulative indicator that adds volume on up days and subtracts volume on down days.

Attribute 2: Price Momentum Oscillator－Movement

Momentum measures the amount that a financial instrument's price has changed over a given timeframe.

Attribute 3: Relative Strength Index－Movement

The RSI is classified as a momentum oscillator, measuring the velocity and magnitude of directional price movements. It is intended to chart the current and historical strength or weakness of a stock or market based on the closing prices of a recent trading period.

Attribute 4: Stochastic-Oscillator

The Stochastic Oscillator is a momentum indicator that shows the location of the close relative to the high-low range over a set number of periods.

Attribute 5: Weighted Moving Average-Movement (WMA-Movement)

A weighted moving average (WMA) has the specific meaning of weights that decrease in arithmetical progression.

Results:

	SVM (RBF)	Logistic	Neural
1-day	43.37%	42.37%	39.56%
5-day	**51.00%**	**49.20%**	**48.19%**
10-day	39.36%	43.78%	40.76%
30-day	43.57%	42.97%	40.56%

Table 2. Initial results from time series data analysis

We observed that using 5-day data returns the best result, and SVM outperformed the other two models in all cases.

3.2 Sentiment Analysis

We decided to use a simple approach, Naive Bayes Classifier, to analyze sentiment in the tweet data set. The Python NLTK provides a built-in Naive Bayes classifier, which can be trained given a labeled feature set and then used to classify future instances.

After training our Naive Bayes classifier on the training set, we ran the classifier on each tweet in our data set. For each tweet, the classier outputs a score in the range of (0, 1). For each day, we computed the average score for all the tweets and used the average score as a benchmark to label each tweet: we labeled the tweets with a score higher than average as positive, and the tweets with a score lower than average as negative. In the end, we divided the number of positive tweets by the total number of tweets within a day to get the daily sentiment score.

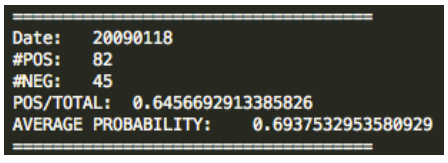

Figure 2. Example output from NLTK Naïve Bayes Classifer

3.3 Combined Analysis

Before incorporating the sentiment scores into original time series data analysis, we labeled the sentiment scores as up, down or

same. We set the cutoffs to 0.52 and 0.56 such that any day with a sentiment score lower than 0.52 is labeled down, between 0.52 and 0.56 is labeled same, and above 0.56 is labeled up.

After adding the labeled sentiment results into the original model, we obtained the results represented in the following table.

	SVM (RBF)	Logistic	NN
1-day	40.21%	44.12%	42.01%
5-day	**51.88%**	**50.42%**	**48.13%**
10-day	43.75%	41.46%	38.54%
30-day	48.54%	46.04%	42.92%

Table 3. Results from combined analysis

From the figures in Table 3, we observed that, in 8 out of the 12 cases, the performance was improved by adding in the sentiment analysis. Overall, there is an average increase of 1.11% in classification correctness among the 12 cases.

The 5-day timeframe still generated the best performances among the four timeframes we considered, and SVM remained to be the best model except in 1-day timeframe.

We also observed that the classification correctness improved the most in 10-day and 30-day timeframes. It can be explained by the fact that, in time series data analysis, the historical data from 10-day or 30-day timeframe was not quite relevant to current-day market movement, and thus the more relevant sentiment results helped to improve the performance more.

3.4 Statistical Tests

We did a one-tailed independent t-test on differences of prediction accuracies of different learning models, different timeframes, and with/without sentiment attribute. We found that 1) Accuracies of using 3 models are not significantly different. 2) 5-day timeframes returned significantly higher accuracies than other timeframes, with critical level between 0.005 and 0.1. 3) Labeled sentiment results help to significantly improve the accuracies of 10-day and 30-day timeframes of using SVM with a critical level of 0.1, but have insignificant impact on other cases.

4. CONCLUSION AND FUTURE WORK

Taking our results and analysis into consideration, we can now answer our three questions posed earlier in the paper.

1. SVM appeared to be the most accurate learning model for predicting market movement. But the statistical tests showed that SVM is not significantly better than logistic regression.
2. Across all three learning methods, 5-days of prior data achieved the highest percentage of correctly classified instances and are statistically better than other timeframes.
3. For most cases, the addition of the Twitter sentiment analysis results appeared to improve performance moderately, and the improvement is only statistically significant in using SVM on 10-day and 30-day of prior data.

For future work, there are three aspects that can be improved on. Firstly, the current data set that we performed sentiment analysis on provided only keywords instead of original tweets. This lack of context may have affected the accuracy of our sentiment analysis. The small training set that we built may be another aspect that can be improved. With only a number of 295 training examples, the learned model is far from being accurate and comprehensive. Thirdly, the sentiment analysis method we used from the Python NLTK is a very simple and preliminary textual sentiment analysis tool. There are a lot of sophisticated tools available in the market that may yield more accurate results.

We believe that with better sentiment analysis tools, a data set providing more contexts for words and a larger training set, it may be possible to further increase the accuracy of the methods presented.

5. REFERENCES

[1] J. Bollen and H. Mao. Twitter mood as a stock market predictor. IEEE Computer, 44(10):91–94. http://arxiv.org/pdf/1010.3003.pdf.

[2] V. H. Shah. 2007. *Machine Learning Techniques for Stock Prediction*. Foundations of Machine Learning, New York University. http://www.vatsals.com/Essays/MachineLearningTechniquesforStockPrediction.pdf.

[3] T. B. Trafalis and H. Ince. *Support vector machine for regression and applications to financial forecasting*. IJCNN2000, 348-353. http://www.svms.org/regression/TrIn00.pdf

[4] H. Yang, L. Chan, and I. King. *Support vector machine regression for volatile stock market prediction*. Proceedings of the Third International Conference on Intelligent Data Engineering and Automated Learning, 2002. http://www.cse.cuhk.edu.hk/~lwchan/papers/ideal2002.pdf

[5] M. Cohen, P. Damiani, S. Durandeu, R. Navas, H. Merlino, E. Fernandez. *Sentiment analysis in microblogging: a practical implementation.* Red de Universidades con Carreras en Informática (RedUNCI), P. 191-200. http://sedici.unlp.edu.ar/bitstream/handle/10915/18642/Documento_completo.pdf?sequence=1

[6] G Garner. *Prediction of Closing Stock Prices.* Course project for Engineering Data Analysis and Modeling at Portland State University, Fall term, 2004. http://web.cecs.pdx.edu/~edam/Reports/2004/Garner.pdf

[7] S. Bird, E Loper, and E Klein. *Natural Language Processing with Python*. O'Reilly Media Inc., 2009. http://ntlk.org/

[8] http://www.infochimps.com/