

Finance Market Prediction: A Hybrid Analysis Model

Amit Kr Tiwari

Under Graduate Student

Department of Computer Science & Engineering, West Bengal University of Technology, India

amiit111tiwari@gmail.com

ABSTRACT

We introduce a novel approach to predict the future of Finance market by applying Machine Learning principles and Sentiment Analysis. In this approach we use the combination of Sentiment Analysis of twitter data and Machine learning approach for historical data, to predict the stock market stability for the next day. Stock prices are considered to be very dynamic and movable to quick changes because of the underlying nature of the stock market and in part because of the mix of known parameters (previous day opening price, previous day closing price etc.) and unknown factors (like Rumors, election etc.). Using machine learning approach we use known parameters for calculating the next day status of finance market and by using sentiment analysis of twitter data we evaluate the finance market status for next day for unknown factors. In the combination of both approach we calculate the stock market value for next day.

Keywords: Naïve Bayes Classifier, Sentiment Analysis, Sentiment Classification, Twitter.

I. INTRODUCTION

Finance Market Prediction trend has long been an interesting topic and is extensively studied by researchers from different fields. Machine Learning, a well-established algorithm in a wide range of applications, has been extensively studied for its potentials in prediction of stock market. Popular algorithms, including support vector machine (SVM) and reinforcement learning, have been reported to be quite effective in tracing the stock market and help maximizing the profit of stock option purchase while keep the risk low [1,2]. We use historical data to predict the upcoming stock value. Efforts have been done to break the boundaries by incorporating external information through fresh financial news or personal interest post such as Twitter. These approaches known as Sentiment Analysis.

In practice, there are two stock prediction methodologies [3]:

Technical Analysis: This method handles the previous or historical data of stock market to predict the future value of the stock market. So, this process is actually based on past pattern. We include known factors in this category like previous day opening data, previous day closing data etc.

Fundamental Analysis: This method is more concern about the current status of the company rather than its market or previous stock value. We include unknown factors in this category like Election, Rumors etc.

When, we apply Machine learning to Stock Data, we are more concerned in doing a technical analysis to see if our algorithm can accurately learn the underlying patterns in the stock time series. As per this process, machine Learning can play a major role on predicting or forecasting the future stock value for any particular company. In this approach, we are not including any unknown factors which may cause a big change for any company's stock market value. We are combining fundamental analysis technique with the technical analysis approach. Now, by using Sentiment Analysis on Twitter data for any particular company, we can get the current status or public opinion about that company which plays a major role in predicting the future stock value for that particular company.

In this project, we are using Naïve Bayes approach for technical analysis. The reason for using this approach is fairly intuitive. Naïve Bayes approaches have been successful in analyzing and predicting phenomena's relying on a time dependence or time series. A Naïve Bayes classifier is trained on a particular stock, where emission probabilities are modeled as Gaussian Mixture Models conditioned on a particular state. The trained model is then used along with the stock values for a certain number of days to forecast the value on the next day. This is done in a Maximum a Posteriori manner, by considering all the possible values of the stock on the $(d+1)^{st}$ day, computing their likelihoods using the trained Naïve Bayes Classifier and choosing the value with the

maximum likelihood. For fundamental analysis part, we are performing sentiment analysis on twitter data. Twitter is a popular microblogging service where users create status

message (which is called “tweets”). These tweets sometimes express opinions about different topics. We are using these tweets to find the people’s sentiment about any particular company. Using Twitter API, we can retrieve the tweets about any particular company for analyzing purpose.

The remaining paper is organized as follows. In section II, we will review some of the existing techniques for finance market prediction. In section III, we will give the detail of our approach which is the combination of Naïve Bayes Classification approach and Sentiment Analysis on Twitter Data. In section IV, we will discuss about result, as per our approach. In section V, we will conclude the paper.

II. Previous Work

Stock Market prediction has attached a great deal of attention in the past. For technical analysis, themachine learning techniques which have been widelyapplied to forecasting stock market data. In this category there are various approaches including Artificial Neural Networks (ANNs) [4], Fuzzy Logic (FL) [5], support Vector Machines (SVMs) [6] and Hidden Markov Model [HMM] [7,8,9,10]. From the available machine learning approaches, ANNs have been very effective and efficient but still there was several limitations with this approach[11]. Wang and Leu proposed an approach in

which they trained a system based on a recurrent neutral network which used features extracted using the Auto Regressive Integrated Moving Average (ARIMA) analyses, which was very effective and also gave reasonable accuracy [12].

III. Algorithm

In our proposed approach, there are two parts on which we are working separately and then at last we combine both the result to find the more accurate data for prediction.

Technical Analysis:

For technical analysis, we are using simple Naïve Bayes Classifier. Efficiency can be improved using other available approaches. [7,8,9,10].

The probability model for a classifier is a conditioned model which can be represented as:

$$P(C|F_1, \dots, F_n) \quad (1)$$

Over a dependent class variable C with a small number of outcomes or classes, conditional on several feature variables F1 through Fn. If the value of n is large, then this model on probability tables is infeasible.

Using Bayes’ theorem, this can be written

$$P(C|F_1, \dots, F_n) = \frac{P(C)P(F_1, \dots, F_n|C)}{P(F_1, \dots, F_n)} \quad (2)$$

Using Bayesian Probability, the above equation can be written as:

$$posterior = \frac{prior \times likelihood}{evidence} \quad (3)$$

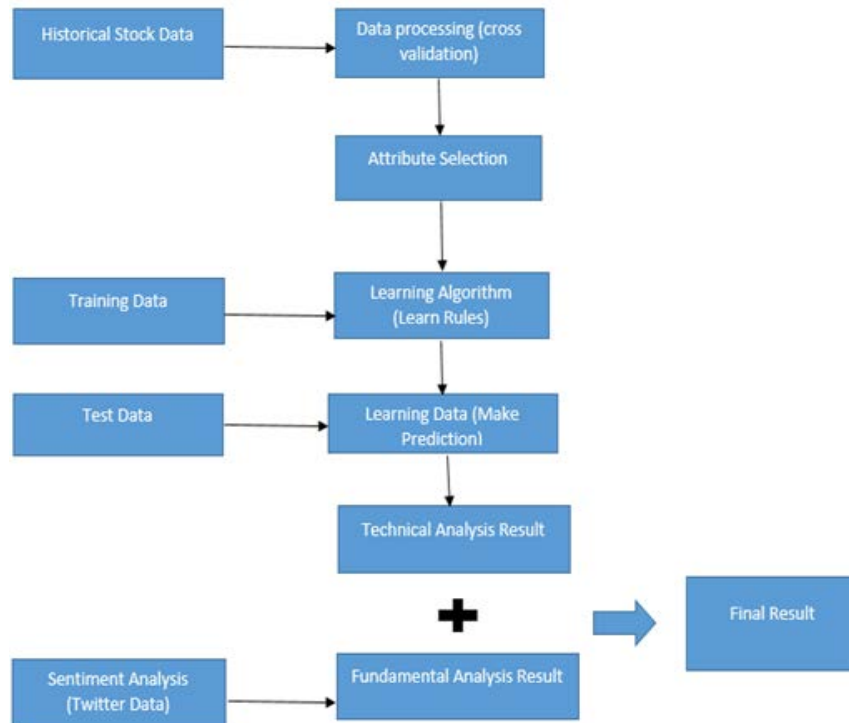


Figure 1: Hybrid Analysis Model Diagram

Fundamental Analysis:

In this part we are using Sentiment Analysis approach to find the current status of any particular company. We are using Naïve Bayes classifier to classify the data.

Naïve Bayes:

Naïve Bayes is a simple model which works well on text categorization [13]. We use a multinomial Naïve Bayes model. Class c^* is assigned to tweet d , where

$$c^* = \operatorname{argmax}_c P_{NB}(c|d)$$

$$P_{NB}(c|d) := \frac{(P(c) \sum_{i=1}^m P(f_i|c)^{n_i(d)})}{P(d)}$$

In this formula, f represents a feature and $n_i(d)$ represents the count of feature f_i found in tweet d . There are a total of m features. Parameters $P(c)$ and $P(f_i|c)$ are obtained through maximum likelihood estimates, and add -1 smoothing is utilized for unseen features.

There are not any large public data sets of Twitter messages with sentiment, so we collect our own data. Twitter has an Application Programming Interface (API) for programmatically accessing tweets by query term. The Twitter API has a parameter that specifies which language to retrieve tweets in. We always set this parameter to English. Thus, our classification will only work on tweets in English because our training data is English-only.

There are multiple emoticons that can express positive emotion and negative emotion. For example, ☺ express positive emotion. In the Twitter API, the query “☺” will return tweets that contain positive emoticons, and the query “☹” will return tweets with negative emoticons. The full list of emoticons can be found in table 1.

For the training data, we use a scraper that queries the Twitter API. Periodically, the scraper sends a query for ☺ and a separate query for ☹ at the same time. This allows us to collect tweets that contain the emoticons listed in Table 1.

The Twitter API has a limit of 100 tweets in a response for any request. The scraper has a parameter that allows us to specify the frequency of polling. We found an interval of 2 minutes is reasonable polling parameters.

The training data is post-processed with the following filters:

1. Emoticons listed in Table 1 are stripped off. This is important for training purposes. If the emoticons are not stripped off, then the few approaches like MaxEnt and SVM classifiers tend to put a large amount of weight on the emoticons, which hurts accuracy.
2. Any tweets containing both positive and negative emoticons are removed. This may happen if a tweet contains two subjects. These tweets are removed because

we do not want positive features marked as part of negative tweet, or negative features marked as part of a positive tweet.

3. Retweets are removed. Retweeting is the process of copying another user's tweet and posting to another account. This usually happens if a user likes another user's tweet. Retweets are commonly abbreviated with “RT.”

4. Repeated tweets are removed. Occasionally, the Twitter API returns duplicate tweets. The scraper compares a tweet to the last 100 tweets. If it matches any, then it discards the tweet. Similar to retweets, duplicates are removed to avoid putting extra weight on any particular tweet.

After post-processing the data, we take the first 800,000 tweets with positive emotions, and 800,000 tweets with negative emoticons, for a total of 1,600,000 training tweets.

Table 1: List of Emoticons

Emoticons Mapped To ☺	Emoticons Mapped To ☹
☺	☹
:)	:(
:D	
=)	

The test data is manually collected, using the web application. A set of 177 negative tweets and 182 positive tweets were manually marked. Not all the test data has emoticons. We use the following process to collect test data:

1. We search the Twitter API with specific queries. These queries are arbitrarily chosen from different domains. For example, these queries consist of companies (Google, Microsoft).
2. We look at the result set for a query. If we see a result that contains a sentiment, we mark it as positive or negative. Thus, this test set is selected independently of the presence of emoticons.

IV. Result

Fig1 represents our Hybrid Analysis Model approach which is using Fundamental Analysis and Technical Analysis result to predict the final stock value of the Stock Market. In this approach we are using Naïve Bayes Classifier which is appropriate in many cases but still not so much efficient in many cases. So, by using another approaches like HMM (Hidden Markov Model), SVM (Support Vector Machine) [7,8,9,10] we can increase the efficiency of Technical Analysis which is based on the historical data. The metric used to evaluate the performance of the algorithm is Mean Absolute Percentage Error (MAPE) in accuracy. MAPE is the

average absolute error between the actual stock values and the predicted stock values in percentage.

$$MAPE = \frac{1}{n} \sum_{i=1}^n \frac{|p_i - a_i|}{a_i} * 100\%$$

Where a_i is the actual stock value, p_i is the predicted stock value on day i and n is the number of days for which the data is tested. Table 2 lists the MAPE values for the six stocks using our approach.

Table 2: MAPE value

Stock Name	Hybrid Model
Apple Inc.	1.463
Microsoft Corp.	1.213
Google	0.982

V. Conclusion

We present hybrid analysis models which combine the effort of Machine Learning and Sentiment Analysis to predict or forecast the future of finance market. We present a Naïve Bayes approach for stock prediction based on the historical data. The model uses a latency of d days to predict the stock value for the $(d+1)^{st}$ day. A MAP decision is made over all the possible values of the stock using a previously trained classifier. For Sentiment Analysis on twitter data we use Naïve Base classifier to classify the tweets. In this approach, we are trying to combine the technical analysis and fundamental analysis methodology to predict the finance market value with more accuracy.

VI. REFERENCE

1. W. Huang et al., "Forecasting stock market movement direction with support vector machine," Computers & Operations Research, 32, pp. 2513-2522, 2005.
2. J. Moody, et al., "Learning to trade via direct reinforcement," IEEE Transactions on Neural Network, vol. 12, no.4, Jul. 2001.
3. Vatsal H. Shah, "Machine learning techniques for stock prediction," www.vatsals.com.
4. J.H. Choi, M.K. Lee, and M.W. Rhee. Trading s&p500 stock index futures using a neural network. Proc of the Third Annual International Conference on Artificial Intelligence Applications on Wall Street, pages 63-72, 1995.
5. L. Cao and F.E.H. Tay. Financial forecasting using support vector machines. Neural Computation and Application, pages 184-192, 2007.
6. E. Vercher, J.D. Bermudez, and J.V. Segura. Fuzzy portfolio optimization under downside risk measures. Fuzzy Sets and Systems, pages 769-782, 2007.
7. B. Nobakht, C.E.J. Dippel, and B. Loni. Stock market analysis and prediction using hidden markov models.
8. L. R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. Proceedings of the IEEE, pages 257-286, 1989.
9. Hidden markov model and the Baum-welch algorithm. IEEE Information theory society newsletter, Dec 2003.
10. A. Gupta, B. Dhingra. Stock Market Prediction using Hidden Markov Models.
11. M.R. Hassan. A combination of hidden markov model and fuzzy model for stock market forecasting. Journal of Neurocomputing, pages 3439-3446, 2009.
12. J.H. Wang and J.Y. Leu. Stock market trend prediction using arima-based neural network. IEEE Int. Conf. on Neural Networks, 4:2160-2165, 1996.
13. C.D. Manning and H. Schutze. Foundations of statistical natural language processing. MIT Press, 1999.