Stock Movements: What Does Twitter Know?

In this project we want to explore the monetization of Twitter data in the stock market. The final goal is to make a profit in the market based on having a one day advantage in predicting the next day movement of Microsoft and Apple stocks. We find that Sentiment Analysis on Twitter data is mature enough and can boost prediction accuracy of current alternatives.

1. Overview

1.1. Introduction
We want to compare the effectiveness of Twitter data mining for investment strategies in the stock market. Trend analysis from social media is a newly established feature of web giants like Facebook and Twitter. However it is not yet clear how to use this tools in algorithmic investment. Traditionally, investors have analysed quantitative data to identify patterns and/or trends that can drive investment decisions. In this scenario it's clear what the data is. For example financial analysts might try to predict the near future by running a simple regression on the price history of the stock of the Apple, the industry index as well as an average index of the market, such as S&P 500. In contrast to such statistical models based on historical financial data, one might try to identify the public sentiment towards Apple and hence predict whether the stock will move up or down. In order to compare the methods as fairly as possible we set up a trading strategy that utilizes a next-day predictor. We then run the strategy using the different predicting methods and rank them according to performance.

1.2. Summary of Results
Due to time limitations we were able to collect 3 months worth of Twitter data. However, we found that Sentiment Analysis is a powerful tool even with this limited training datasets. Our prediction algorithms based on Twitter outperformed the ones trained on financial data. In particular Deep Neural Networks on 3 months data had more than 80% accuracy whereas an SVC classifier trained on 4 years worth of financial signals performed only 63%. According to our analysis, sentiment recognition from Twitter data is a good predictor for stock movements.

1.3. Organization
In Section 2 you can find a brief data description. In Section 3 and 4 we describe the sentiment analysis methods and the financial data driven predictor respectively. Our trading algorithm is described in Section 5 and our findings are summarized in Section 6. We conclude this report with future work in section 7.
2. Data

For this project we use two sources of data, Yahoo stock closing quotes and Twitter messages. Choosing the right data is as crucial as developing right algorithms. Hence, some research papers use arbitrarily selected historical data [1] that often introduces bias and raises questions about the authors motives to select a particular time period. This the main reason why we decide to use current data with all ongoing events. In this setting we have no control of the marker or events.

One popular source of public opinion is twitter[2]. It is social media web service where users post short messages on random topics. The data is freely available for research use. However, twitter limits data distribution license that forced us to collect our own data. We launched a server to collect Apple and Microsoft related tweets. Since twitter api has download quota per user, every three seconds we searched twitter for “Apple” and “Microsoft” keywords and saved the results. We collected over 300Mb of data over three months period that came down to 56 days excluding weekends and holidays.

Daily stock quotes were much easier to obtain. Yahoo finance provides main daily indices such as opening, closing, and average stock price. In our experiments we use four years of closing price. Also, we try combining two sources of data resulting in only 56 data points. Since this is extremely small amount of data, in all experiment we use 20 fold cross validation to evaluate our algorithms.

3. Sentiment Analysis and Natural Language Processing

3.1. NLP features
In recent years NLP has brought many advanced techniques for text processing. While the performance still far from perfect, it is coming close to what is expected from a human. This is why NLP is becoming a popular tool in predicting stock price movements.

Various techniques were applied in recent years by researcher to predict stock movements. One of the most popular (and the most simple) is to use of unigrams [3,4]. This is simple word count, often normalized. The idea behind this approach is that some words might express correlation with market. In our experiments we found that unigrams did not perform well.

Some other features that we use are 1) word counts - total number of words used with a relation to the company, 2) tweet count, 3) “buzz” correlation - how many times a tweet was retweeted. While 2 and 3 features showed some improvement, they were far behind our best technique and we decide to leave them out.
3.2. Using Twitter data for stock prediction
Social media, as the collective form of individual opinions and emotions, has very profound though maybe subtle relationship with social events. This is particularly true when it comes to public Tweets and stock trading. In fact, research has shown that when it comes to financial decisions, people are significantly driven by emotions. These emotions, together with people's opinions, are in real-time reflected by tweets. As a result, by analyzing relevant tweets using proper machine learning algorithms, one could grasp the public's sentiment as well as attitude towards the stock's price of interest, which could intuitively predict the next move of it.

Some previous work has been done to show that tweets can indeed reflect stock price change. Randomly selected three months' tweets, and pointed out that, surprisingly, the ‘Calmness’ score of these tweets is able to resemble some of the key features on Dow Jones Industrial Average (DJIA) price change within the same period of time. Other work focused on one single stock and used particular person’s tweets to predict that stock’s price change. However, there is not yet any published work aimed to predict any single stock's price using machine learning through social media, like tweets.

3.3. Word based sentiment
Sentiment analysis has shown to have some correlation with stock quotes [3,6]. We use AFINN word based sentiment dictionary [5]. This is a dictionary of words where each word has assigned score. For example a word “good” might have a score “+2”, “awesome +5” and “terrible -5.” We evaluate each word, sum up all the scores together, and normalize by the word count in each tweet. This is done to avoid larger sentiment scores in longer tweets. After doing all this we plotted the data to see if there exist a correlation. The plots also contain quadratic fit function to show the trends. To our surprise quotes and sentiment have high correlation. This is a promising feature to rely on for our experiments. Plotted results can be found in the Figure 3.1 and 3.2.
3.4. DNN sentiment

3.4.1 Motivations
In section 3.3 we already make some sentimental analysis about words. However, the words based sentiment analysis might not be accurate in some cases. Language itself are complex, in one single sentence, people usually try to include different kinds sentiment about different things. What make modeling sentences sentiment even harder is sometime people’s opinions might be expressed in a very subtle way. In this case, if we only use bag of words model like we did in section 3.3, it is really hard to guarantee that we included very words which have sentimental opinions in our dictionary before we obtain the testing data.

Furthermore, language itself are involved with complex structure. For example: “Yesterday's weather is horrible, but it does not bother me to enjoy that movie.” This example obviously give a positive sentiment about the movie but negative to the weather. If we only use bag of words based model. There is no way the model could figure out which sentiment refers to which object.

In this case, we need to take the structure information into our consideration for sentences modeling. In order to capture all the sentiment for each word (despite its magnitude of the sentiment) and the instinct tree structure for a given sentence, we propose to use convolutional neural networks in our project.
3.4.2 Model

The model architecture, shown in Figure 3.3 is a slight variant of the CNN architecture of collobert. The sentence can be represented as following:

\[ x_{1:n} = x_1 \oplus x_2 \oplus \ldots \oplus x_n, \quad (1) \]

each \( x \) means one word from sentence. and plus circle sign mean the concatenation operator. In general, \( X_{\{i:j\}} \) refer to the concatenation of words from \( i \) to \( j \). A convolution operation involves a filter \( W \), which is applied to a window of \( h \) words to produce a new feature. For example, a feature \( c_i \) is generated from a window of words \( X_{\{x:i+h-1\}} \) by

\[ c_i = f (W \cdot x_{i:i+h-1} + b). \quad (2) \]

\( b \) is a bias term and \( f \) is a non-linear function such as the hyperbolic tangent. This filter is applied to each possible window of words in the sentence to produce a feature map:

\[ c = [c_1, c_2, \ldots, c_{n-h+1}], \quad (3) \]

then we applied a max-over-time pooling operation over the feature map and take the maximum value as the feature corresponding to this particular filter. The idea behind this max-pooling is to capture the most important feature.

In order to capture enough variance of the feature, we randomly initialize 100 different filters to generate 100 dimensional feature representation for each sentence. Then we treat this representation as input to deep neural networks to train the filters and feature maps.
4. Finance Data Analysis

4.1 Motivation
Our aim is to set up our comparison of methods as fairly as possible. That's why we want to feed our financial data into a model that targets the same or a closely related learning problem as the sentiment analysis methods. We want to underplay the long-term financial trends and focus on picking up the daily differences. Stocks of tech giants like Apple and Microsoft have rallied over the last 5 years, following an appreciation of the technological sector in total (Figure 4.1).

![Figure 4.1 Apple and Microsoft Stock Price for 2010-2015.](image)

However this is of little use in predicting whether Apple will go up or down tomorrow. Our machine should be able to predict the next day movement of the stock. Hence we have to collect data at a more granular level. Things like daily returns and volatility indicators will be essential in our efforts to achieve a good learning curve.

4.2 Model
For our financial analysis we chose to use a Support Vector Classifier. We experimented with different kernels but showed that Linear Kernel performed the best for both stocks. We trained our classifier on approximately 1500 days of data and tested in on approximately 250 days. The challenging part of our project is choosing the right features.

What we want is to push long term trends to the background. Hence we will try to stay as close as possible to a Markovian model. A Markov process has the property of predicting the future based on the current state only. Thus our stochastic model should be based on a Geometric Brownian Motion which means that we will be looking at the behavior of , i.e. the returns. Daily returns do a much better job than price, in obeying the Markovian principles. For intuition notice the absence of long-term trends in the daily return graph (Figure 4.2), contrary to the long term price indices we saw above (Figure 4.1).
Despite the fact that past returns cannot reflect future ones, trends do exist and past price behaviors influence future prices. Therefore we will take today's returns to forecast next day's. As our targets are Tech stocks, we want to add 'IXCO', basket made of technical sector stocks. Sector price may react to events related to that specific industry and directly impact AAPL and MSFT. UST represents (widely) interest rate movements, it is in fact the IR corresponding to 7-10 years US Treasury bonds. It is suggested that high yields / low prices are bad news for stocks: first of all, corporate bonds prices follow, the result being higher financing costs thus lower profits. A corollary being investors sell stocks to buy bonds for a higher returns and/or a lower risk. Finally VIX, the Volatility Index is generally correlated with uncertainty on the markets, and translates into a bearish trend when volatilities are high. Also VIX volatilities are implied which means their represent a forecast (1day, 1 week, 1 month) of a general tendency.

Figure 4.2 Apple and Microsoft Daily Returns for 2010-2015.

4.3 Data
It makes sense to use features that are correlated to the behavior of the stock, nevertheless it is important to carefully select them. Too many features result means too much noise and uncertainty. Too few features will give too much bias. Using subject knowledge and experimentation we addressed the variance-bias tradeoff by selecting four features: Stock Daily Returns (AAPL), Sector Price Index (IXCO), Interest Rate (UST), Volatility Index (^VIX). Our target data is the direction of the stock the next day. In our data we also keep track of the next day return difference (Diff). However the only use of this column is to take its sign in order to create the target value next day movement (color): 1 means up and -1 means down (Figure 4.3).
5. Trading Strategy

To verify how predictions, even not 100% accurate, can be profitable, we played backtestings with two trading strategies: one using our predictor, the other not using it. We wanted a strategy relying on trends rather than immediate price movement, because we wanted to include trading costs, to make things more realistic. In fact, a strategy that buy and sell at random has a lot of chances to capture the global movement of the prices on the time period and therefore might not be representative. Penalizing random trades with brokerage costs helps highlighting the ‘real’ profitability of a strategy. The baseline strategy looks at the proximity of the price to the MA 3 days. We consider there is a trade reversal when price crosses up or down the 3 days MA. (This comes from the 3 ticks rule of thumb that suggests a trend is established after three identical movements). Baseline strategy has been played on AAPL stock on the last 100 days of the test set. Some constraints are added: limited capital and limited short positions. A risk constraint also has been added which is to flat a short position as soon as possible.

6. Results

6.1. Prediction Accuracy
Using 20 fold cross validation we construct four experiments for each company. The first one is simple linear regression, this is our baseline. Then we experiment with SVM model using financial data as the input. The third experiment is based on SVM model and NLP based twitter features. Lastly, we use DNN model to train and predict twitter sentiment that is used as input to SVM model.
The baseline, simple linear regression, for both companies produces 53% accuracy so does financial data based SVM with accuracy of 60%. For Microsoft Word based sentiment and text features produced 57% and 80% accuracy is achieved with DNN sentiment. For Apple predictions are slightly better. Text based features predict 75% of stock movement and 83% DNN sentiment. All results can be found on Figure 6.1.

![Figure 6.1 - prediction accuracy for Microsoft and Apple](image)

6.2. Trading strategy
The baseline strategy is not really profitable: for $100,000 invested, the net realized is $-900 corresponding to the brokerage fees.

![Graph showing trading strategy](image)

As we can see on the above graph, the strategy keeps buying when prices raise and keeps...
selling on the downside. Adding a predictor signaling a reversal may help. The second strategy will buy on the upside too but only if the predictor says so. Then the position is held and sold when the combined (MA + predictor) gives a Sell signal. Same thing on the opposite side (but with limited exposure to 100 short stocks)

As shown in the figure above, this strategy successfully captured the raise that occurred in February 2015. More generally the strategy seems to be adequate on big market movements, and also reduce the risk and the fees by generating much less transactions than the baseline. Result is not bad: +12%, a net result of $12000 for 100 days of trading.

7. Future Work

Due to limited time constraints we were not able to try and implement all techniques. The next steps in this project are to expend information extraction domain from twitter to news and other social media. Also, more advanced word representations are known to be helpful. The final goal that was planned from the beginning is to set up online automated system to learn and predict stock prices.

8. References

[1] Huang et al., “Forecasting stock market movement direction with support vector machine.”
CSc 84010 Final Project “Stock Movements: What Does Twitter Know?”