Twitter Cashtags and Sentiment Analysis in Predicting Stock Price Movements

Chee, Kwuan H.

June 7, 2017

Abstract

Despite the Efficient Market Hypothesis, stock market prediction is an area of interest for many researchers and investors. There are many observed variables which impact the movement of stock prices such as financial reports and economic policy. However, the expected behavior of the market may not always be the actual outcome, thus there are some variables that are not captured initially unaccounted for. We believe one of these variables is sentiment. More specifically, investor sentiment towards a stock or a company will affect their trading decisions, thus affecting the stock price. By gathering large amounts of data on individuals' moods, we can aggregate this data and interpret it as public sentiment. There are several possible sources of public sentiment information, however given the expressive nature of social media participants and the growing amounts of data available via social media, we can use it as a source to extract public sentiment knowledge. One social media in which the users are more expressive is Twitter. Although much work has been done on predicting stock price movements based on general public sentiment data on Twitter, we would like to further investigate this topic. We will investigate whether or not including a cashtag attribute, which indicates whether or not a Tweet contains a cashtag or not, in our model leads to better prediction of the daily DJIA change. The cashtag is essentially a stock ticker symbol and is implemented and used in a very similar way to hashtags. We will build two models, one with the cashtag attribute and one without and we will compare them. This will provide an indication of whether or not Tweets with cashtags are more beneficial to the overall performance of stock price movement prediction.

Contents

1 Introduction

1.1 Overview and Motivation

Despite research done in this area was initially based on the random walk theory and the Efficient Market Hypothesis [1] [4]. Stock market prediction has been extremely popular recently and the topic attracts people from various fields. According to the EMH, stock market prices are largely driven by new information and follow a random walk pattern and therefore cannot be predicted by past information. However, the EMH is problematic in two ways: firstly, several studies indicate stock market prices do not follow a random walk and can be predicted to a certain extent [7] [9] [13]. Secondly, although news may be unpredictable, there are early indicators that can be extracted from social media in order to predict changes in economic and commercial sectors. As a result, researchers still try to tackle this problem as many are interested in determining if patterns can be found using machine learning techniques to predict stock price movement, a more general application of accurate stock prediction is maximizing returns on stock investments.

In the financial sector, there is a community of investors that utilizes knowledge of the markets, data and economic theory in order to develop models to somewhat predict stock price movement and outperform the market. The methods used by large financial companies to predict and make buy/sell decisions take into account many variables and constantly receive new data. While there are many factors that influence a stock's price on any given day, one variable of interest recently is public sentiment. We know from research in psychology that the emotion and moods of individuals influence their decision making process [5]. Behavioral economics has also provided further evidence that emotion and moods are the driving force behind financial decisions [12]. Therefore, in addition to their knowledge of the market, it is likely that these investors respond to general sentiment towards companies and stocks, which in turn leads to fluctuations in stock prices. What if we had a system in place that tried to understand these same things?

In order to develop a model for predicting stock prices, we could simply acquire stock market performance data and public sentiment data which we could compile and run machine learning algorithms, which would provide us with our desired model. For our stock market performance variable, we will likely use the daily opening price change and we can acquire day to day stock price data from Google Finance. For our public sentiment data, there are several sources such as newspaper articles, blog posts, and social media. Conversely, we could use expert knowledge or sentiment acquired from several handpicked sources, however doing this would not be helpful in answering our research question as this would not be representative of the public. Consequently, we need to extract this information from a more general source, such as social media, in order to get a more accurate representation of public sentiment.

Social media is often used as a platform by people for freely expressing their thoughts and opinions, one service often used solely for this purpose is Twitter. While we recognize that each individual Tweet may not necessarily provide valuable information in predicting market movement, the aggregation of vast amounts of Tweets can provide valuable insight on public sentiment. The aggregation of sentiment data from many individuals' Tweets provides us with knowledge of the public mood. We will use this public sentiment data along with data on daily opening stock price changes to find patterns between the two in order to develop a predictive model which can be used for predicting stock price direction in the future.

While there is plenty of existing work on predicting the stock market using Twitter data and sentiment analysis, we wanted to diverge slightly and investigate We've noticed that there isn't any work that investigates whether or not including a cashtag attribute in their model would help lead to better stock price prediction. We hypothesize that Tweets containing cashtags may be too positive or negative about stocks. By including the cashtag attribute in our model, we may enable the machine learning algorithm to discover this bias if it exists and discount or account for the effects of Tweets containing cashtags. One concern we have with existing work is that many researchers often prune their data, removing periods of extreme volatility in the stock market in order to improve the predictive power of their models. However, this is unrealistic because volatile periods exist. Therefore, we will test our model on a data set which has not been pruned in order to see how applicable our model is in the real world.

2 Background and Related Work

There is a comprehensive collection of related work that investigates predicting stock price movement using data obtained from Twitter. This work contains many different implementations in the preprocessing of Twitter Data, different sentiment analysis techniques and different machine learning techniques in order to predict stock market price movement. Rao and Srivastava [14] modeled a set of causative relationships between search volume index data from Google and Twitter sentiment data to model movements in Oil, Gold, foreign exchange markets and market indices such as the Dow Jones Industrial Average (DJIA) and the NASDAQ-100. They also investigate the lagged causative effect of Tweet sentiment produced during active market days and inactive days as well as the search behavior of the public before price changes. Rao and Srivastava [14] used a naive classifier to classify incoming Tweets as either positive or negative Tweets. For sentiment analysis of the Tweets, they used a lexicon/JSON API from Stanford NLP research group's Twitter sentiment. In order to gather search volume data, they collected weekly search volumes for search terms related to the respective sectors. Once they had their data, they performed a Granger Causality Analysis to further test the causality of SVI and Tweet sentiment on market securities by using it to identify correlation patterns across several time series at different lagged intervals. Next, they needed to build a model to predict stock market. They used Expert Model Mining System (EMMS), Auto Regressive Integrated Moving Average (ARIMA) and seasonal ARIMA models. They chose these methods as they are used commonly in financial modeling to predict the value of stocks. In the end they were able to correctly predict the weekly direction (gain or loss) of the DJIA at a rate of 94.3%.

Bollen and Mao [3] investigated whether measurements of mood states derived from large-scale Twitter feeds are correlated to the value of the DJIA over time. They used OpinionFinder (OF) and Google-Profile of Mood States (GPOMS) to measure the sentiment of their Tweets. In order to ensure that OF and GPOMS are able to accurately represent various aspects of public mood, they cross-validate the resulting mood series, comparing their ability to detect the public's response during Thanksgiving and the presidential election in 2008. Although both OF and GPOMS are able to identify the public response on those days, they found that certain GPOMS moods partially overlapped with OF moods. Therefore, GPOMS provided a more distinct perspective on public moods unattainable by the uni-dimensional OF. Similar to Rao and Srivastava [14], Bollen and Mao [3] also carried out a Granger causality analysis to find out whether a lagged X variable (mood) will exhibit significant correlation with Y (DJIA prices). Although their Granger Causality analysis suggests a predictive relationship between mood and DJIA prices, the relationship between these two variables is not linear, but the Granger causality analysis is based on a linear model. As a result, they propose using a Self-Organizing Fuzzy Neural Network (SOFNN) in order to better address the non-linearity between the two variables. Their SOFNN takes two sets of inputs: (1) the past 3 days of DJIA values, and (2) the same combined with permutations of their mood time series data (past n days of mood data). They ultimately use the SOFNN as their predictive model and in order to measure its forecasting accuracy, they use the average Mean Absolute Percentage Error (MAPE) and the direction accuracy (increase or decrease in DJIA price). Bollen and Mao [3] was able to achieve a directional accuracy of 87.6%

Bing et al. [2] proposed a method in order to determine if the stock prices of specific companies are more predictable as well as whether a selection of 30 companies listed in the NASDAQ and the NYSE can be predicted by the 15 million records of Twitter data they have, which are all related to the companies of interest. In order to do this, they considered extracted trivial textual Tweet data through the use of NLP techniques. More specifically, they considered each instance of a Tweet as a combination of several words and phrases and they classified the Tweets sentiment into one of five categories: Positive+, Positive, Neutral, Negative and Negative-. Afterwards, they apply chi-square test and adjusted residual to identify the relevant patterns between public sentiment and stock market prices. This was used to find association rules between the attributes. They also implemented three other classic data mining algorithms (Naive Bayers classifier, support vector machine (SVM) and C4.5) to compare the predictive accuracy with their proposed algorithm. They were able to achieve an average directional predictive accuracy of 76.12% with their proposed algorithm.

Makrehchi et al. [10] proposed a novel approach by manually labeling social media text using significant stock market events. They use two approaches for mood and sentiment detection. The first is lexicon-based mood detection, which is considered an unsupervised learning approach. This approach involves calculating presence and counts of chosen words or phrases in the document. The second approach is also the proposed approach and is based on supervised learning. Generally, supervised learning garners better results, however it is impractical for certain use cases due to a lack of labeled data. As a result, they propose to extract automated labels from the stock market by monitoring market performance and finding out which days are good or bad for the market. With the collected training data containing Tweets labeled with positive and negative sentiments, they are able to build a classifier and predict sentiments of Tweets. The text that needs to be labeled can be either pre-event, post event and current (at the time) according to the use case. Ultimately, their underlying model predicts the sentiment of future Tweets, the labels (sentiment) are then used to make a prediction for the stock market. Makrehchi et al. [10] was able to beat the S&P 500 by roughly 20% in returns in four months.

Vu et al. [15] aimed to predict the daily directional movements of four tech companies (Apple, Google, Microsoft and Amazon) stock prices. Their proposed model combined the positive and negative sentiment of Tweets, consumer confidence in a product according to ′bullish′ or ′bearish′ lexicon as well as the past three days stock market movement. Their approach diverges from the traditional sentiment analysis for tracking the DJIA by Bollen and Mao [3] as they don't pre-assign a sentiment lexicon or assume mood dimensions. Instead, as briefly mentioned previously, they induce the lexicon automatically by association with ″bullish″ (long gains) and ″bearish″ (short gains) anchor words on the internet. They obtained their Tweets using the Twitter Firehose API and built a named entity recognition (NER) system on Twitter data to identify and remove ′noisy′ Tweets. Noisy Tweets were ones which included one of their queries, but was completely unrelated in their context, e.g. a query of ′mac′ which results in a Tweet about mac and cheese. The NER implemented was based on a linear Conditional Random Fields (CRF) model which was used to detect whether Tweet contained name entities related to the companies. The entities they focused on were persons, organizations, hardware and software. In order to determine what these were, the researchers manually labeled 3665 randomly sampled Tweets related to the companies of interest, they were able to garner 540 unique entities. Their features were used in a Decision Tree classifier and using cross-fold validation they were able to yield accuracies of 82.93%, 80.49%, 75.61% and 75% in predicting the directional movements of Apple, Google, Microsoft and Amazon, respectively.

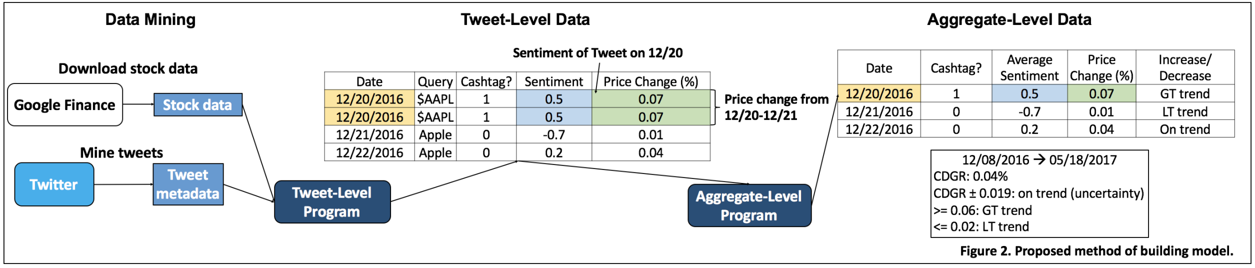
Mao et al. [11] worked to determine whether there was a correlation between the number of Tweets that mention the Standard & Poor 500 (S&P500) and two S&P500 stock indicators-the stock price and daily traded volume. They examined this at three levels-they first examined it at the stock market level and how the number of Tweets mentioning the S&P500 is correlated to price changes and trade volume of S&P500 stocks. Then, at the industry sector level, they look at the correlation between the number of Tweets mentioning the ten Global Industry Classification Standard (GICS) and the stock price and daily traded volume of the respective sectors. Finally, they analyzed this from the individual stock level, in particular, they focused on the correlation between the volume of Tweets mentioning Apple Inc.'s stock with its daily price change and daily traded volume. In order to do this, Mao et al. [11] applied a linear regression with Twitter data as exogenous input in order to predict the closing price and daily traded volume. Upon establishing a correlation exists between Twitter predictors and stock market indicators, Mao et al. [11] wanted to test the predictive power of their model. They used 38 days of data as the training set data for building their predictive model. In the end, they were able to achieve an accuracy of 68% in predicting the directional change in daily S&P500 stock price on the stock market level. On the industry sector level, they also achieved an accuracy of 68%, but in predicting the directional change in the daily traded volume of S&P500 companies in the Financials sector. Finally, on the stock market level, they achieved a 52% accuracy in predicting daily directional change in daily traded volume of Apple stock.

Hentschel and Alonso [8] conducted a study of cashtags on Twitter, however he did not investigate sentiment analysis and instead presented an analysis of cashtags on Twitter. In their research, they studied the distribution of cashtags, the characteristics of users Tweeting with cashtags, the relationship between cashtags and between cashtags and hashtags and they study whether there is a connection between Tweet performance (volume of cashtag Tweets) and market performance. One aspect of the relationship between cashtags which Hentschel and Alonso [8] examined was the co-occurrences of cashtags. With this they built a co-occurrence graph where the nodes represented the cashtags and the edges represented co-occurrences between the cashtags. From this co-occurrence graph they were able to affirm existing knowledge such as Microsoft being close competitors with Google and Apple as well as interesting new knowledge, in their case they found that Vringo was in a lawsuit with Google. In their attempts to determine a relationship between Tweet volume and the stock price, they found that a correlation only exists sometimes and indicate that sentiment analysis may play an important role.

3 Our Approach

As outlined above, much of the existing work that uses data from Twitter to predict stock market movements follows a common general approach. In this general approach, the data is mined to extract features that are thought to predict stock market movements, those features are used in training a statistical model, and the model is then used to make predictions. The approaches differ primarily in the data features the researchers focus on. The approach we will implement follows from the hypothesis that public sentiment may be a predictive attribute of market movements since our moods and emotions impact our decision making. Therefore, the approach requires that we gather public sentiment data along with a market performance variable and compile it so we can run machine learning algorithms in order to develop the models we need to answer our research question. The machine learning approach we will be using is an instance-based supervised learning approach, where we provide the correct label for each instance's class in the training set for the machine learning algorithm to learn and predict in the future. Each instance is characterized by the values of the included attributes which measure a different aspect of the instance. While we could follow the common general approach in a similar fashion as much of the existing work, we wanted to investigate a different aspect of the topic of stock market prediction. We initially queried some companies listed on the Dow Jones on Twitter and it was clear that there are two distinct types of Tweets. There were Tweets with stock ticker symbols, e.g. $AAPL, $WMT, etc. and there were also Tweets just mentioning the company name, e.g. Apple, Wal-Mart, etc. Upon this discovery, we want to investigate whether including a cashtag attribute which indicates whether a Tweet contains a cashtag or not within the general approach will improve the overall accuracy of the model.

Figure 1: Proposed method of building model.



Date	Query	Cashtag?	Sentiment	Price Change (%)
12/20/2016	$AAPL	1	0.5	0.07
12/20/2016	$AAPL	1	0.5	0.07
12/21/2016	Apple	0	-0.7	0.01
12/22/2016	Apple	0	0.2	0.04

Date	Cashtag?	Average Sentiment	Price Change (%)	Increase/ Decrease
12/20/2016	1	0.5	0.07	GT trend
12/21/2016	0	-0.7	0.01	LT trend
12/22/2016	0	0.2	0.04	On trend

12/08/2016 → 05/18/2017
CDGR: 0.04%
CDGR ± 0.019: on trend (uncertainty)
>= 0.06: GT trend
<= 0.02: LT trend

Figure 2. Proposed method of building model.

4 Methods and Design

4.1 Data Mining

Given the overall approach outlined, we would like to evaluate it and follow it in helping us answer our research question. More specifically, we've followed our proposed approach and developed a model using Tweet data and stock price data. In order to begin, we needed to acquire the necessary data to answer our research question. Firstly, for our market performance variable, we chose to use the daily DJIA opening price change. We chose to investigate an index rather than an individual company due to the relatively stable nature of an index compared to a company. We calculated the daily price change using daily DJIA opening price data over the period from December 8th, 2016 to May 18th, 2017 obtained from Google Finance. One small issue we had was that the market is closed on the weekends so we would be unable to obtain price change data for the day Friday through Sunday. In order to resolve this, we simply just removed all weekend instances. We also mined roughly two million Tweets over the same period of December 8th, 2016 to May 18th, 2017 using the Twitter API. We chose Twitter because its API is simple to use, vast of amounts of data exists on it and people freely use it as a platform in order to express their thoughts and opinions. The format of the mined Tweet data was in JSON and we used a Python library called TinyDB to store and access the data later on. Since we want to determine whether including the cashtag attribute in our model would provide it previously unknown information about the sentiment of cashtag Tweets that would lead to a more accurate model. We needed to ensure that we had instances of both cashtag and non-cashtag Tweets for our model. So, we would want to mine Tweets containing the name of companies listed on the DJIA as well as Tweets containing the corresponding ticker symbol or cashtag of that company. Therefore, we mined two types of Tweets:

- Tweets that contain the name of a company listed on the DJIA, e.g. JPMorgan Chase.
- Tweets that contain cashtags, the corresponding stock ticker symbol of a company listed in the DJIA, prefixed by a dollar sign, e.g. $JPM.

4.2 Tweet-Level Data

Once we've acquired the necessary raw Tweet data and stock price data, we input it into a program we wrote which extracts the sentiment of each Tweet and ultimately collates the data so that the Tweet date and price change date aligns. We also use sentiment analysis in order to extract sentiment information from each Tweet. Sentiment analysis is the process of computationally identifying and categorizing opinions expressed in a piece of text as negative or positive, often expressed with a value between -1 and 1. The sentiment analyzer we use is part of a Python library called TextBlob. TextBlob's sentiment analyzer is based on the Pattern library (a web mining module for Python) and achieves an accuracy of 75% for movie reviews. We use TextBlob's sentiment analyzer because it is easy to implement given previous time constraints and it produces relatively accurate results. It is important to note that we process Tweets including cashtags slightly before we calculate its sentiment. Upon the realization that cashtag Tweets were primarily composed of cashtags and that our sentiment analyzer scores each word (based on a lexicon) then does math to provide the sentiment polarity, we decided to remove all cashtags from cashtag Tweets.

Once we strip the cashtags from the Tweet, we calculate its sentiment from the remaining words. We also use the raw Tweet data to label each Tweet as containing a cashtag (1) or not (0). The program outputs a file where each row is an instance representing a Tweet posted on date t, with the sentiment score of that Tweet, a 1 or 0 indicating whether or not the Tweet contains a cashtag, as well as the price change from date t to t+1. While we can build a model using this data, these instances don't capture what we're trying to model in order to answer our research question. Each of these instances is representative of a single Tweet, thus, the sentiment of a single Tweet, however what we're trying to capture is public sentiment. If we included individual instances each representing a single Tweet, we would be trying to determine how individual sentiment impacts market movement. The resulting model would not be helpful in answering my research question and would likely be inaccurate given the confusion the machine algorithm will face from variability in sentiment between individual Tweets.

4.3 Aggregate-Level Data

In order to resolve the issue of each instance representing the sentiment of a single Tweet's, we aggregate the sentiment of all Tweets on one day into a single row so each instance now captures public sentiment rather than just the sentiment of a single Tweet. We do this by taking our Tweet-level data and inputting it into our aggregate-level program which aggregates the sentiment and labels each instance's class. We also normalize for the natural upward trend of the Dow Jones. Since the Dow Jones, like most other indices have a natural uptrend, there are some days where the daily price change is more positive or negative than the trend and we wanted to capture this detail in the model. We use the data to calculate the compounded daily growth rate over the observed period and this value is used to assign a label to each instance's class. Now each row of the aggregate-level data represents an instance with the average sentiment of a query on date t, the price change from date t to t + 1, a 1 or a 0 indicating whether the Tweets used contained a cashtag or not as well as the labeled class.

4.4 Aggregate-Level Data Row Explanation

The five columns of data are: the date, a 1 or 0 indicating whether the Tweets used to obtain the average sentiment contained a cashtag or not, the average sentiment, the DJIA opening price change (%) and a class indicating whether the daily price change was greater than, less than or on trend. The date is the date being observed-only Tweets and stock data from that date will be used for the average sentiment. The cashtag? attribute indicates whether the Tweets used in calculating the average sentiment of that row included only cashtag Tweets (cashtag? = 1) or only non-cashtag Tweets (cashtag? = 0). The average sentiment on each row represents the average sentiment of all the Tweets on the same date mentioning a specific company. The price change is the daily percentage change in the DJIA index from the Tweet date, t, to the next day, t + 1. Finally, the last column is our class and it is what we're attempting to predict once we've built our model. We initially only included two options in the class: increase or decrease based on whether the price change value was positive or negative. But, to normalize for the natural upward trend of the DJIA, the class is now based on the comparison between the daily price change and the compounded daily growth rate of the Dow Jones index over the observed period. There are three options in the class now: on trend, greater than trend and less than trend. Although we added another label, when we built our model we did not have single instance in our dataset which was labeled on trend. As a result, our model was only able to classify instances as greater than trend or less than trend.

4.5 Removing Social Robots

One problem we faced was robots, it was quite evident that social robots are a problem, not just within Twitter, but all of social media. In fact, there was recently a case where social bots were involved in manipulating the stock market. A company called Cynk deployed an army of social bots to incite chatter on social media about the company products, which led to this company's stock to spike, ultimately making a $6 billion profit. Nonetheless, this was a problem which needed to be addressed since it appeared that accounts which appeared to be robots (e.g. sometimes Twitter handle is something like stockbot) were sending out many Tweets which contained cashtags. We hypothesize that the Tweets sent out by these bots add noise to the sentiment data and the overall model as bot sentiment is not representative of human sentiment. If bot accounts are sending out the majority of the Tweets with cashtags, then the cashtag attribute would no longer only be capturing whether Tweets contained a cashtag or not, but also indirectly whether the Tweet was posted by a bot or not. Therefore, if we are using data which includes robot spam to build a model in trying to determine sentiment this may be problematic. We need to identify and remove Tweets by robots. The variety of tools available for detecting Twitterbot accounts is rather limited. In fact, the BotOrNot Python API appears to be one of the only available options for this problem. Thus, we decided that the BotOrNot Python API would be our solution for bot detection. Their bot detection works given an account name and evaluates whether the account is controlled by a human or a machine based on user data scraped from the user account name. Davis et al. [6] created BotOrNot, which has a classification system which generates over 1,000 features using available meta-data and information extracted from interaction patterns and content. They've grouped these features into 6 main classes: Network, User, Friends, Temporal, Content and Sentiment features. Each of these classes that captures the class used in describing are analyzed and used in determining whether an account is operated by a human or a bot. Once BotOrNot runs its algorithm on an account, it returns a numeric value from 0 to 1 indicating the 'bot likelihood' of the account, which represents how likely it is that the account is a bot. Since we don't want to include Tweets sent by bots in our data as it would lead to a misrepresentation of the public sentiment, we need to determine a suitable bot likelihood value at which we want to remove a Tweet. To do this, we looked at many examples of Tweets, sent by both known bot and non-bot accounts and then we calculated the bot likelihood score using their API. We wanted to find a value which would maximize the number of correctly classify users that are bots. After looking at many accounts, the bot likelihood score which classified an user as a robot we decided on was 0.55. However, when trying to implement this to check through all the Tweets we've mined, we encountered several problems with the Twitter API Rate Limit. One of our initial solutions was to create a list of bot users and another list of non bot users based on the bot check. The list would be used in later iterations to check whether or not we've already checked the bot status of the user. This allows us to reduce the number of requests we will be making to the Twitter API as the number of unique users shouldn't be too high.

4.6 Building the Model

In investigating the research question, we initially built a couple of models using learning algorithms such as the Naive Bayes classifier and the J48 Decision Tree classifier in order to possibly gather some preliminary information about the dataset. These models did not provide us with much new insight in regards to the dataset or classify the instances with a satisfactory level of accuracy. Nonetheless, the machine learning algorithm we have chosen to use to build the model is Instance Based-k or IBk. Both the Naive Bayes classifier and the J48 Decision Tree classifier performed statistically significantly worse than IBk in terms of the percentage of correctly classified instances. IBk classifier implements the k-nearest neighbors (k-NN) classification algorithm. It works by classifying objects based on a majority vote of its neighbors, with objects assigned to the class most common among its k-nearest neighbors. The value of k is often a small, positive integer. We decided the value of k by using a meta learning algorithm in Weka called Cross Validation Parameter Selection. We used it to do a cross validation on the same dataset with different values of k between 2 and 50 in order to optimize the value of k which yields the model with the highest number of correctly classified instances. The value of k was always set equal to 2. It is important to note that we carry out some data preprocessing before supplying the data to the model. Firstly, we resampled the dataset so that each class has an even number of instances. We also create two different dataset files, one which includes the cashtag attribute and one which doesn't. We will use these two different datasets in order to determine the significance of the cashtag attribute.

5 Results

The results we've obtained use Twitter and stock data obtained over the period from December 8th, 2016 to May 18th, 2017. In order to obtain these results, we built two models one with a cashtag attribute and one without and both using IBk learning algorithm. The model without the cashtag attribute only slightly outperforms the model with cashtags in terms of the percentage of total correctly classified instances, achieving 1.7% greater accuracy. Although, we resampled our dataset and we only have two labels, greater than trend and less than trend, so our baseline for the two models is 50%. Given this, both the models overall perform roughly 15% above the baseline. Another immediate observation is that both models do a far better job at classifying instances that are greater than trend versus less than trend. We hypothesize that this misclassification of instances into less than trend may be a result of our sentiment analyzer being unable to fully capture negative sentiment on Twitter. More specifically, sarcastic and negative Tweets which would be difficult for most analyzers to pick up on. Finally, we conducted a corrected paired t-test on the percentage of total correctly classified instances in order to determine the significance of the cashtag attribute. The results of the t-test indicate that while there was a small difference, the difference wasn't statistically significant.

6 Future Work

Upon reevaluating the overall model we've deliberated on several ways of improving the overall results. Firstly, while we recognized that Tweets posted by Twitter bots are a problem as they produce extra noise in the aggregation of sentiment data, we were unable to implement bot detection. In the future, we should investigate either removing Tweets by accounts we consider to be bots either through the BotOrNot API by figuring out a fix to the API Rate limit constraints or by developing our own bot detection algorithm.

Furthermore, the machine learning algorithms we initially used as well as IBk are fairly simple algorithms which produce fairly accurate results, but not the best. In the future, we could investigate using more complex machine algorithms more suited for this type of classification such as neural networks, random forests and support vector machines.

Another improvement which could be made would be to evaluate and include financial variables or attributes which are currently used in existing models in order to help gain an edge on the stock market. While our research question aimed to investigate the inclusion of the cashtag attribute, if our true goal was to maximize the models ability to predict the DJIA daily directional movement we should try to provide the model with as much relevant information as possible.

Another important attribute is the number of followers of the Tweet's author. We hypothesize that Tweets sent out by users with a higher average number of followers will have a greater impact on public sentiment as the sentiment in their Tweet may impact the sentiment of their followers. It will be interesting to see if inclusion of a follower attribute which provides information regarding the number of followers in the model would lead to an improved model.

One large and fairly basic improvement we could make towards improving our model is with the sentiment analyzer we use. We rely on a Python library called TextBlob and while it achieves a 75% accuracy on movie reviews, this number is likely lower for Tweets. Thus, a suitable next step could be trying to build or train an existing sentiment analyzer just using Tweets. Given the unique characteristic of Tweets, they are 140 characters or less, it is likely that a classifier trained on Tweets will produce more accurate sentiment data.

References

[1] Eugene F et al. Fama. "The Adjustment of Stock Prices to New Information". In: *International Economic Review* 10.1 (1969), pp. 1–21. URL: http://EconPapers.repec.org/RePEc:ier:iecrev:v:10:y:1969:i:1:p:1-21.

[2] Li Bing, Keith C. C. Chan, and Carol Ou. "Public Sentiment Analysis in Twitter Data for Prediction of a Company's Stock Price Movements". In: *Proceedings of the 2014 IEEE 11th International Conference on e-Business Engineering*. ICEBE '14. Washington, DC, USA: IEEE Computer Society, 2014, pp. 232–239. ISBN: 978-1-4799-6563-2. DOI: 10.1109/ICEBE.2014.47. URL: http://dx.doi.org/10.1109/ICEBE.2014.47.

[3] Johan Bollen and Huina Mao. "Twitter Mood As a Stock Market Predictor". In: *Computer* 44.10 (Oct. 2011), pp. 91–94. ISSN: 0018-9162. DOI: 10.1109/MC.2011.323. URL: http://dx.doi.org/10.1109/MC.2011.323.

[4] Paul H Cootner and Paul Cootner. *The random character of stock market prices*. English. Originally published: [Cambridge, Mass.] : MIT, 1964. London : Risk, 2001. ISBN: 1899332847.

[5] Amos Tversky Daniel Kahneman. "Prospect Theory: An Analysis of Decision under Risk". In: *Econometrica* 47.2 (1979), pp. 263–291. ISSN: 00129682, 14680262. URL: http://www.jstor.org/stable/1914185.

[6] Clayton Allen Davis et al. “BotOrNot: A System to Evaluate Social Bots”. In: *Proceedings of the 25th International Conference Companion on World Wide Web*. WWW ’16 Companion. Montr&#233;al, Qu&#233;bec, Canada: International World Wide Web Conferences Steering Committee, 2016, pp. 273–274. ISBN: 978-1-4503-4144-8. DOI: 10.1145/2872518.2889302. URL: https://doi.org/10.1145/2872518.2889302.

[7] Liam Gallagher and Mark Taylor. “Permanent and Temporary Components of Stock Prices: Evidence from Assessing Macroeconomic Shocks”. In: *Southern Economic Journal* 69.2 (2002), pp. 345–362. URL: http://EconPapers.repec.org/RePEc:sej:ancoec:v:69:2:y:2002:p:345-362.

[8] Martin Hentschel and Omar Alonso. “Follow the money: A study of cashtags on Twitter”. In: *First Monday* 19.8 (2014). ISSN: 13960466. URL: http://firstmonday.org/ojs/index.php/fm/article/view/5385.

[9] Manolis G Kavussanos and Everton Dockery. “A multivariate test for stock market efficiency: the case of ASE”. In: *Applied Financial Economics* 11.5 (2001), pp. 573–579.

[10] Masoud Makrehchi, Sameena Shah, and Wenhui Liao. “Stock Prediction Using Event-Based Sentiment Analysis”. In: *Proceedings of the 2013 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT) - Volume 01*. WI-IAT ’13. Washington, DC, USA: IEEE Computer Society, 2013, pp. 337–342. ISBN: 978-0-7695-5145-6. DOI: 10.1109/WI-IAT.2013.48. URL: http://dx.doi.org/10.1109/WI-IAT.2013.48.

[11] Yuexin Mao et al. “Correlating S&#38;P 500 Stocks with Twitter Data”. In: *Proceedings of the First ACM International Workshop on Hot Topics on Interdisciplinary Social Networks Research*. HotSocial ’12. Beijing, China: ACM, 2012, pp. 69–72. ISBN: 978-1-4503-1549-4. DOI: 10.1145/2392622.2392634. URL: http://doi.acm.org/10.1145/2392622.2392634.

[12] John R Nofsinger. “Social mood and financial economics”. In: *The Journal of Behavioral Finance* 6.3 (2005), pp. 144–160.

[13] Bo Qian and Khaled Rasheed. “Stock market prediction with multiple classifiers”. In: *Applied Intelligence* 26.1 (2007), pp. 25–33. ISSN: 1573-7497. DOI: 10.1007/s10489-006-0001-7. URL: http://dx.doi.org/10.1007/s10489-006-0001-7.

[14] Tushar Rao and Saket Srivastava. “Analyzing Stock Market Movements Using Twitter Sentiment Analysis”. In: *Proceedings of the 2012 International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2012)*. ASONAM ’12. Washington, DC, USA: IEEE Computer Society, 2012,

pp. 119–123. ISBN: 978-0-7695-4799-2. DOI: 10.1109/ASONAM.2012.30. URL: http://dx.doi.org/10.1109/ASONAM.2012.30.

[15] Tien-Thanh Vu et al. “An experiment in integrating sentiment features for tech stock prediction in Twitter”. In: *Proceedings of the Workshop on Information Extraction and Entity Analytics on Social Media Data*. Mumbai, India: The COLING 2012 Organizing Committee, Dec. 2012, pp. 23–38. URL: http://oro.open.ac.uk/40125/.