Twitter Sentiment on Trump and its Effect on Financial Markets

**M.A.C. van Dijk (724171)**

**June 2017**

**Tilburg University: BSc Economie en Bedrijfseconomie**

**Graduation Date: August 2017**

**Supervisor: prof. dr. E.C.M. van der Heijden**

**Words: 4579**

**Abstract**

During recent years, machine learning techniques have found its way into a multitude of research. This paper attempts to utilize these techniques by analyzing Twitter sentiment surrounding the 45th president of the United States: Donald J. Trump. Researchers, predominantly in the financial field, have taken great interest in analyzing the effects of tweets stemming from Trump on financial markets. The paper at hand will attempt to go beyond analyzing tweets that originate from Trump himself, and will analyze all tweets concerning the United States’ President. The general sentiment in tweets surrounding Trump will be taken as a proxy for measuring overall confidence in this President, with the goal of examining the relationship between this proxy and the Standard & Poor’s 500 index returns. It is concluded that no significant links exist between the proxy and the Standard & Poor’s 500 index. More research is needed to thoroughly substantiate this claim.

*Table of Contents*

1. Introduction……………………………………………….
2. Method……………………………………………………
3. Results…………………………………………………….
4. Discussion………………………………………………....
5. Conclusion………………………………………………...
6. References………………………………………………...
7. Introduction

For the last couple of years, the presence of big-data, social media and machine learning techniques have been a considerably big topic of discussion in business as well as academia. The growth of big-data is expected to continue following a hyperbolic trend and will have increased 300-fold through the years 2005-2020 (Gantz & Reinsel, 2012). With this amount of data and the uprising of social networking sites (hereinafter referred to as SNSs) such as Twitter, Facebook and Instagram, it is likely that valuable data is hidden within these SNSs. Since these SNSs provide such vast amounts of data it only seems logical that they might yield interesting insights into various other fields of research, such as political science (Sandner, Sprenger, Tumasjan & Welpe, 2010), disaster detection (Matsuo, Okazaki & Sakaki, 2010), sentiment analysis (Pak & Paroubek, 2010) and finance (Bollen, Mao & Zeng, 2011).

This paper will attempt to use one of these SNSs (Twitter), and combine it with the field of financial research, in particular for stock market prediction. Earlier research such as Fama (1965) relied heavily upon the Efficient Market Hypothesis (EMH). The EMH states that stock prices are merely driven by new information, implicitly stating that past information should not yield predictive power over future stock returns. Moreover, since news is future information and cannot be predicted, the accuracy of predicting stock price movements cannot exceed 50%. The EMH is supported later on by Samuelson (1965) and proven in the empirics by many papers, as summarized by Jensen (1978).

However, we know by now that the EMH does not hold true in all cases, and that some past indicators do hold statistically significant predictive power over future returns (as summarized by Fama & French, 1992). A case can therefore be made against the preconceived notion, laid upon by the EMH, that the future cannot be predicted by past information. Several papers in the domain of financial research (such as e.g., Basu, 1977; Lo & MacKinlay, 1999) have found empirical evidence to the contrary of what the EMH predicts. Moreover, looking more closely in the field of research that this paper will lay focus on, several other papers (such as e.g., Bartov, Faurel, & Mohanram, 2016; Bollen, Mao, & Zeng, 2011) have found statistically significant predictive power over stock returns with the help of sentiment analysis via Twitter. Furthermore, textual analysis and machine-learning techniques can be applied to different fields of financial research as well. Kearney and Liu (2014) summarize how textual analysis has predictive power in many fields of financial research such as predicting revenues, trading volumes of stocks and stock returns. Their summary shows how the analysis of written text in various media outlets has significant predictive power over important financial measures of companies and aggregated markets alike.

This paper however is not another textual analysis model being applied to the stock market or to some financial measures of companies. Expanding on the existing literature, this paper will attempt to explore the relationship between overall Twitter sentiment concentrated around a specific keyword. This keyword will be the 45th U.S. President, Donald J. Trump. Since the 45th U.S. Presidential elections have taken place not very long ago, academia has yet to catch up on researching the effects of such an active Twitter user (averaging 18.4 tweets per day) *and* U.S. President on the stock markets. Moreover, at the writing of this paper, there has not been done any research yet regarding the effects of the tweets sent by Trump. Additionally, no research has been conducted on tweets that are sent with respect to him. Hence, investigating the textual sentiment surrounding these tweets might yield valuable insights for financial research. This paper will not research what effect tweets sent by Trump himself will have on stock markets, but rather what the effect is of the overall sentiment regarding the U.S. President. In a sense, the tweets sent on Twitter regarding Trump will be taken as a proxy to measure sentiment surrounding the U.S. President and thus implicitly the financial markets. While we know that textual sentiment on Twitter surrounding specific stocks has predictive power over the returns of said stocks (Bartov, Faurel, & Mohanram, 2016), no prior research has been done into the effects of sentiment on markets regarding such a specific keyword. The research question of this paper can therefore be defined as follows: Do links exist between sentiment surrounding Trump and stock market returns?

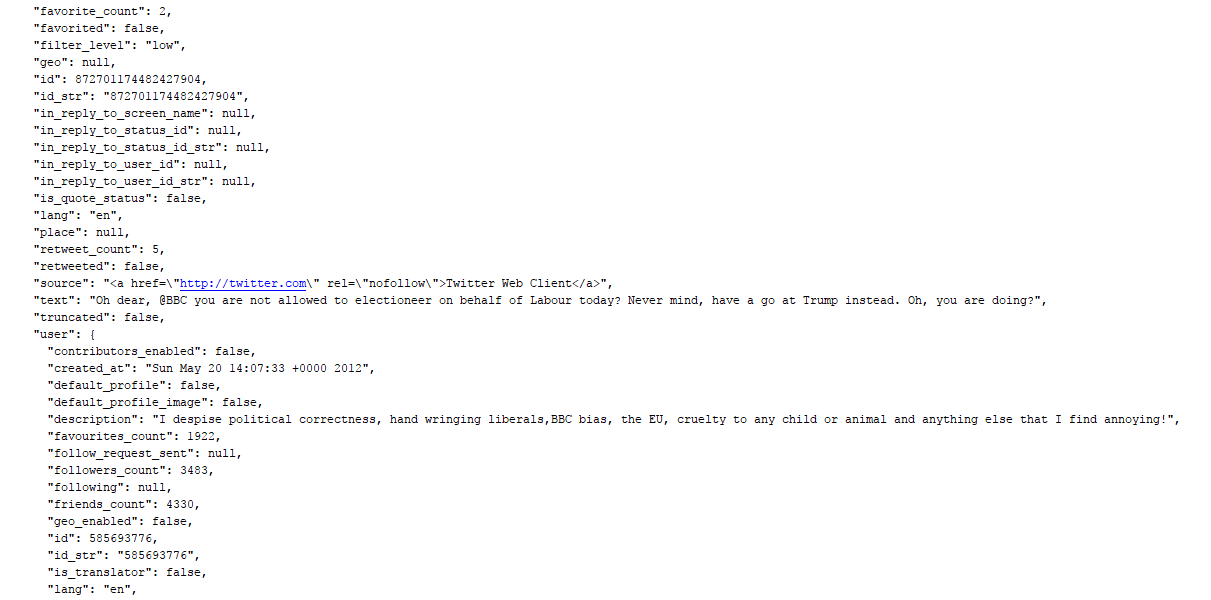
In order to analyze the sentiment of Tweets, this paper will make use of the Twitter Streaming API[[1]](#footnote-1) to load in tweets at real-time speed. Afterwards, a machine-learning algorithm will analyze tweets and classify them as either negative, positive or neutral. The polarity measure induced from said tweets, defined as the number of positive tweets divided by the number of negative tweets, will be put next to the Standard & Poor’s 500 index (hereinafter referred to as S&P500) returns.

The results indicate that no significant links exist between the proxy of market confidence and the S&P500 returns.

The paper is structured as follows. Section 2 explains the methodology used for cleaning, processing, analyzing and storing tweets. Section 3 displays the results of the study. Section 4 discusses opportunities for improvement and gives recommendations for future research. Section 5 concludes.

1. Method

Tweets are obtained in real-time via the Twitter Streaming API, with the help of the Python (3.5) programming language. Within this environment, good use is made of the Tweepy module, in order to keep the program that is written by the author of this paper as accessible as possible. The program is expected to run every trading day from one hour before market open in the U.S. to one hour after market closure. Starting the program means manually inserting the run-time and additionally the interval to save the data. This has to be done manually for each trading day, if not there will be no data for a given day *k*. When being loaded in, a tweet looks like the following.



1

3

2

Figure 1. An example of a tweet being loaded in by the Twitter Streaming API, including most metadata. (1), (2) and (3) are used in the process of cleaning and analyzing the tweet. Redundant metadata not used in this paper is not shown.

Within this figure, number 1 indicates the language the tweet is sent in, number 2 indicates the text of the tweet as it would be seen on Twitter itself and number 3 indicates the number of followers the sender of this tweet has.

While this paper will make most use of the “text” portion of a tweet to analyze sentiment, the metadata as provided by Twitter makes it considerably easier to clean and filter tweets. Hence, before tweets will be analyzed they run through a filtering process that is based on the one being used by Bollen, Mao and Zeng (2011). Furthermore, in addition to the filtering techniques applied by said authors, this paper will not take into account any tweets that are not written in English, tweets that mention a YouTube link or tweets that originate from persons that have less than 50 followers. These additions help to ensure that the sentiment analysis is done only on those tweets that show significant value to the overall sentiment within a given time-period *t*. Moreover, they make sure that the sentiment of tweets is not biased towards any direction due to an overflow of, for example, retweets or YouTube links.

Tweets are classified using machine-learning algorithms that have been previously trained on a subset (corpus) of tweets. During any given day *k*, there will be two streams of tweets running simultaneously, with each of these streams having their own algorithm. One of these algorithms is provided for free by the TextBlob[[2]](#footnote-2) module, this algorithm is partially based on the one as described by Narayanan, Arora and Bhatia (2013). It is pre-trained on an IMDB movie review corpus and makes use of Naïve Bayes Classification[[3]](#footnote-3) for sentiment extraction. Implementing what types of sentiment the classifier recognizes is simply done by labelling the data according to what you would like it to recognize. In the case of this paper, the classifier is trained to recognize positive, negative and neutral sentiment. The second algorithm used is also a Naïve Bayes Classifier algorithm that is trained by the author of this paper on a corpus of tweets as provided by the NLTK library for Python[[4]](#footnote-4). The tweets within this corpus are already labeled as either positive or negative by the creator of said corpus. Making use of the Naïve Bayes Classifier algorithm is fairly easy, since it can be directly loaded in by the same Textblob module that includes the pre-trained classifier. All that has to be done is feeding this algorithm a labelled dataset, which in the case of this paper are positively or negatively labelled tweets. Eventually, the classifier had been fed 5000 tweets to train on, of which 2500 were positively labelled and 2500 were negatively labelled. In order to ensure that the classifiers worked as was intended, they have both been subject to testing before any analysis was conducted on the actual data. A total of 2000 tweets have been used for testing purposes, of which 1000 were positively labelled and 1000 were negatively labelled. Deducing from these tests it was found that the Textblob classifier correctly predicted sentiment in 87.35% of cases, while the classifier that was trained on tweets itself resulted in an accuracy score of 77.65%. While initial expectations were that the Textblob classifier would yield worse results, as it had been trained on movie reviews instead of tweets, it performed substantially better. This could be attributed to the fact that the number of tweets used for training purposes was too small, leading to bias in the classification of tweets. Nevertheless, it was decided to include both classifiers for further analysis, in order to check for consistency of the final results.

Since the program is expected to run live during every trading day, sentiment is being measured as tweets are being streamed in. The program uses a trinary distinction in mood, measuring the amount of positive, negative or neutral tweets that are being streamed in. An example of tweets and classification can be seen below.

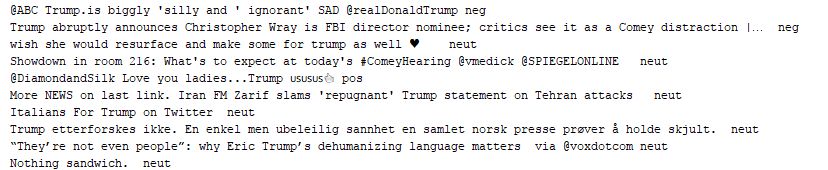


Figure 2. Example of the pre-trained Textblob classification algorithm. On the left: Text of the tweet. On the right: Classification of the respective tweet.

Whenever a tweet is labelled as positive, the program will increase the variable ‘*pos\_count’* to *pos\_count + 1.* The same goes for the variable ‘*neg\_count’* that keeps track of all negative tweets. Neutral tweets however will not be saved into a variable, as they do not provide any contribution to the measure of overall sentiment (Cao, Duan & Yu, 2013). Every *t* minutes, these sentiment variables are reset and a polarity measure is deduced from them (note that *t* = 5 in this paper).

Polarity here is simply defined as:

*Polarityt =*

In essence, this measure describes how positive or negative the sentiment surrounding Trump is at any given time *t*. Both *pos\_count* and *neg\_count* could, in theory, take any value greater than or equal to zero. The polarity measure induced from dividing the number of positive tweets by the number of negative tweets results in what is called the polarity measure. Higher values of this polarity measure indicate a more positive sentiment surrounding Trump, while lower values indicate the opposite.

When *t* minutes have past, the program will append the polarity measure to another variable that keeps track of all past polarity measures for a given day *k*. Furthermore, every *t* minutes the S&P500 index data will be pulled in from the Google Finance API. This data will then be saved to a separate variable for an entire day *k.* Additionally, a time-key will be added to the data such that specific events that might happen during the day can be linked to the polarity and/or S&P500 data within a given time period. At the ending of each trading day, all of the measures that have been saved during the day will be put together into one (Excel) file for storage and easy access for further analysis. Hence, at the ending of each day the data is structured as follows.

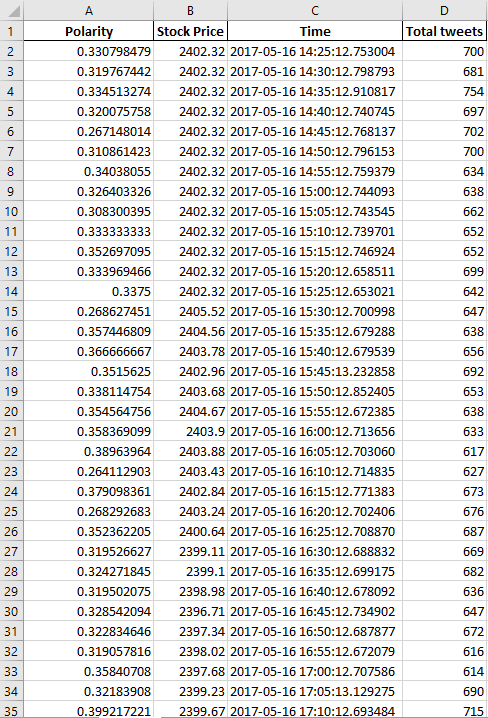
****

Figure 3. Data shown after it has been cleaned, processed, analyzed and saved to an Excel file by the Python script.

Figure 4. Data shown after it has been cleaned, processed, analyzed and saved to an Excel file by the Python script.

1. Results

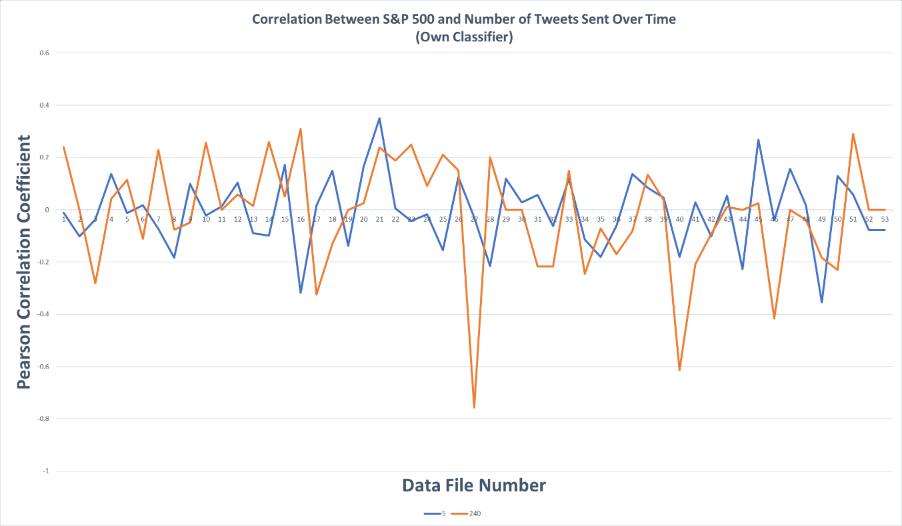
At the ending of the data collection period, 53 trading days were analyzed, which equates to about three months of data as markets do not open in weekends and on special holidays. Additionally, some days were missed due to issues with the program not starting up. Data collection happened between 22-02-2017 and 29-05-2017. Eventually, a total of 2,580,672 tweets were analyzed between both classifiers, meaning that 47,790 tweets were analyzed per day on average. The greatest recorded jump in the value of the S&P500 happened on 10-03-2017, where the index opened 0.43% higher. The greatest recorded decrease in value happened on 27-03-2017, where the S&P500 fell by about 0.88% when markets opened. While this information is interesting on its own, the main goal of this paper is to see whether sentiment surrounding Trump positively or negatively affects market sentiment, and thus the returns of the S&P500.

In order to start analyzing how the measure of polarity covaries with the S&P500 index, it is important to normalize the data that is obtained. In this way, it is ensured that the data remains comparable. This paper uses a simple formula to normalize the initial data within Excel. Additional columns were added in order to ensure that the data would not be overridden. The normalization procedure can be described as follows.

*=*

Where indicates either the price of the S&P500, the measure of polarity or the total number of tweets sent. *t – 5* is used since data is only collected and stored every five minutes. In essence, this formula indicates how a variable changes relative to its previous state. When applying this formula across variables, it will lead to better comparability of said variables.

Since the main concern of this study is to investigate how the S&P500 covaries with keyword polarity, Pearson correlation coefficients are analyzed, including statistical tests to see whether these correlation coefficients are statistically significantly different from zero. Additionally, it is not only interesting to examine covariance between the S&P500 and keyword polarity, but also the covariance between the S&P500 and the total number of tweets sent. It is known from past research that the number of tweets sent in itself might have predictive value over future returns as well (Asur & Huberman, 2010). Furthermore, the focal point herein is observing a statistically significant effect between past changes in independent variables (i.e. the measure of polarity or number of tweets sent) affecting present values of the dependent variable (i.e. S&P500 price), as this would enable for a profitable trading strategy. Hence, before further analysis, lagged independent variables were introduced, up to a maximum lag of 240 minutes (4 hours). In order to visualize how this data is stored at the ending of the recording period, a graph has been included that displays how the correlation coefficients move over time. Only correlations concerning the S&P500 returns and the number of tweets sent over time are taken into account. Furthermore, only the first and the last lagged correlations are taken into consideration, as adding in all lags would clutter the graph substantially. The results of this visualization are as follows.



4

1

2

3

Figure 5. Pearson correlation coefficients over time for the first and last lag. Systemic events that happened during the recording period are marked on the top-right graph. (1) Trump’s tax cut announcement aftermath. (2) Trump’s healthcare bill fails to pass house. (3) First and second round of the French elections. (4) Trump’s budget plan announcement.

First, a look is taken at the co-movements between the number of tweets sent and returns of the S&P500. In order to do so, it is examined whether the Pearson correlation coefficient is statistically different from zero, for all of the lagged variables. This analysis cannot be done by implementing t-tests for the data, as it is expected that the data is not normally distributed around the sample mean (see appendix for proof). Therefore, Wilcoxon signed-rank tests (Wilcoxon, 1945) will be imposed to test whether the data is statistically different from zero. The hypothesis that the sample means for the Pearson correlation coefficients are symmetrically distributed around zero (H0) is tested against the hypothesis that the sample means do not follow a symmetrical distribution around zero (H1). Furthermore, it is analyzed in which direction the variables interact with each other by taking an average of the Pearson correlation coefficients across all file. Taking this average is done by summing the correlation coefficients per lag, across all files, and then dividing the outcome by the sample size of 53. The analysis is conducted separately for both the number of tweets sent and the polarity measure. The results are portrayed on the next page.

|  |  |  |  |
| --- | --- | --- | --- |
| Lag/Market | Standard & Poor's 500 | Lag/Market | Standard & Poor's 500 |
| 5 | -0.0070  (657) | 125 | -0.0251  (495) |
| 10 | -0.0142  (608) | 130 | 0.0419  (443)\* |
| 15 | -0.0031  (676) | 135 | -0.0133  (569) |
| 20 | 0.0175  (552) | 140 | 0.0100  (598) |
| 25 | 0.0095  (595) | 145 | -0.0209  (584) |
| 30 | 0.0001  (527) | 150 | -0.0190  (560) |
| 35 | -0.0101  (680) | 155 | 0.0460  (442) |
| 40 | 0.0396  (613) | 160 | -0.0049  (621) |
| 45 | -0.0312  (522) | 165 | -0.0333  (531) |
| 50 | -0.0047  (631) | 170 | 0.0443  (520) |
| 55 | -0.0280  (520) | 175 | 0.0069  (600) |
| 60 | 0.0258  (493) | 180 | 0.0535  (446)\* |
| 65 | -0.0029  (649) | 185 | -0.0551  (424)\*\* |
| 70 | -0.0171  (578) | 190 | 0.0280  (589) |
| 75 | -0.0189  (588) | 195 | -0.0343  (574) |
| 80 | 0.0014  (652) | 200 | -0.0019  (559) |
| 85 | -0.0188  (561) | 205 | 0.0174  (548) |
| 90 | 0.0552  (391)\*\* | 210 | -0.0155  (489) |
| 95 | -0.0311  (578) | 215 | -0.0117  (531) |
| 100 | 0.0257  (661) | 220 | 0.0101  (465) |
| 105 | -0.0411  (494) | 225 | -0.0358  (377) |
| 110 | 0.0596  (421)\*\* | 230 | 0.0264  (490) |
| 115 | -0.0051  (606) | 235 | -0.0430  (413) |
| 120 | 0.0096  (547) | 240 | -0.0214  (452) |

Table 2. Average Pearson Correlation coefficients between the S&P500 returns and the number of tweets sent, at a given lag in minutes. Uses the sentiment classification algorithm from the Textblob module for Python. Wilcoxon’s W-statistic in parentheses. \*\*\* p < .01; \*\* p < .05; \* p < .10

Table 1. Average Pearson Correlation coefficients between the S&P500 returns and the number of tweets sent, at a given lag in minutes. Uses the sentiment classification algorithm as trained by the author of the paper. Wilcoxon’s W-statistic in parentheses. \*\*\* p < .01; \*\* p < .05; \* p < .10

Average Pearson Correlation Coefficients Between S&P 500 and Number of Tweets (Own classifier)

Average Pearson Correlation Coefficients Between S&P 500 and Number of Tweets (Textblob classifier)

|  |  |  |  |
| --- | --- | --- | --- |
| Lag/Market | Standard & Poor's 500 | Lag/Market | Standard & Poor's 500 |
| 5 | -0.0045  (670) | 125 | -0.0226  (616) |
| 10 | 0.0226  (536) | 130 | 0.0015  (611) |
| 15 | 0.0092  (625) | 135 | -0.0080  (649) |
| 20 | -0.0118  (636) | 140 | 0.0428  (497) |
| 25 | 0.0059  (681) | 145 | 0.0075  (562) |
| 30 | -0.0129  (572) | 150 | -0.0606  (439)\*\* |
| 35 | 0.0274  (603) | 155 | 0.0199  (611) |
| 40 | -0.0182  (661) | 160 | 0.0160  (639) |
| 45 | 0.0153  (633) | 165 | -0.0255  (542) |
| 50 | 0.0050  (681) | 170 | -0.0068  (622) |
| 55 | -0.0106  (638) | 175 | 0.0341  (489) |
| 60 | 0.0094  (681) | 180 | 0.0091  (557) |
| 65 | -0.0096  (661) | 185 | -0.0666  (486)\* |
| 70 | 0.0178  (611) | 190 | 0.0134  (660) |
| 75 | -0.0161  (551) | 195 | 0.0448  (587) |
| 80 | -0.0015  (639) | 200 | 0.0268  (554) |
| 85 | -0.0084  (655) | 205 | 0.0043  (576) |
| 90 | -0.0123  (623) | 210 | -0.0006  (572) |
| 95 | 0.0312  (523) | 215 | 0.0833  (353)\*\* |
| 100 | -0.0081  (644) | 220 | -0.0639  (360)\*\* |
| 105 | 0.0124  (612) | 225 | -0.0057  (534) |
| 110 | 0.0001  (684) | 230 | 0.0354  (457) |
| 115 | 0.0113  (558) | 235 | 0.0046  (479) |
| 120 | -0.0182  (600) | 240 | -0.0380  (457) |

|  |  |  |  |
| --- | --- | --- | --- |
| Lag/Market | Standard & Poor's 500 | Lag/Market | Standard & Poor's 500 |
| 5 | 0.0067  (640) | 125 | 0.0332  (493) |
| 10 | -0.0258  (592) | 130 | -0.0112  (631) |
| 15 | 0.0417  (458)\*\* | 135 | 0.0423  (461)\* |
| 20 | 0.0037  (556) | 140 | -0.0244  (524) |
| 25 | -0.0234  (523) | 145 | 0.0029  (655) |
| 30 | 0.0205  (627) | 150 | 0.0221  (590) |
| 35 | -0.0465  (513) | 155 | -0.0303  (501) |
| 40 | 0.0464  (494)\* | 160 | -0.0144  (615) |
| 45 | -0.0210  (545) | 165 | -0.0020  (616) |
| 50 | -0.0229  (570) | 170 | 0.0345  (495) |
| 55 | 0.0442  (413)\*\* | 175 | -0.0078  (645) |
| 60 | -0.0206  (611) | 180 | 0.0306  (553) |
| 65 | 0.0183  (547) | 185 | -0.0281  (580) |
| 70 | 0.0180  (623) | 190 | -0.0318  (531) |
| 75 | -0.0120  (655) | 195 | 0.0580  (463)\* |
| 80 | -0.0008  (636) | 200 | -0.0128  (538) |
| 85 | 0.0140  (683) | 205 | 0.0636  (490) |
| 90 | 0.0038  (638) | 210 | 0.0187  (500) |
| 95 | 0.0103  (659) | 215 | -0.0242  (506) |
| 100 | 0.0209  (601) | 220 | 0.0154  (536) |
| 105 | -0.0114  (631) | 225 | -0.0377  (448) |
| 110 | -0.0318  (610) | 230 | 0.0171  (534) |
| 115 | -0.0027  (682) | 235 | 0.0287  (497) |
| 120 | -0.0069  (605) | 240 | 0.0356  (425) |

|  |  |  |  |
| --- | --- | --- | --- |
| Lag/Market | Standard & Poor's 500 | Lag/Market | Standard & Poor's 500 |
| 5 | -0.0101  (676) | 125 | 0.0141  (566) |
| 10 | -0.0275  (513) | 130 | -0.0218  (535) |
| 15 | -0.0003  (650) | 135 | 0.0011  (629) |
| 20 | 0.0185  (576) | 140 | 0.0023  (589) |
| 25 | -0.0106  (645) | 145 | -0.0083  (591) |
| 30 | 0.0357  (484)\* | 150 | -0.0139  (587) |
| 35 | 0.0106  (631) | 155 | 0.0051  (610) |
| 40 | -0.0398  (458)\*\* | 160 | 0.0151  (618) |
| 45 | 0.0121  (637) | 165 | -0.0365  (550) |
| 50 | 0.0029  (640) | 170 | 0.0129  (603) |
| 55 | 0.0066  (643) | 175 | 0.0239  (521) |
| 60 | 0.0339  (450)\*\* | 180 | -0.0480  (534) |
| 65 | -0.0086  (600) | 185 | 0.0509  (554) |
| 70 | 0.0171  (604) | 190 | -0.0398  (553) |
| 75 | -0.0164  (613) | 195 | 0.0685  (468) |
| 80 | -0.0208  (522) | 200 | -0.0116  (513) |
| 85 | 0.0208  (508) | 205 | -0.0824  (399)\* |
| 90 | -0.0140  (532) | 210 | 0.0057  (440) |
| 95 | 0.0211  (563) | 215 | -0.0065  (510) |
| 100 | 0.0172  (598) | 220 | -0.0054  (492) |
| 105 | -0.0408  (440)\*\* | 225 | 0.0128  (484) |
| 110 | -0.0222  (641) | 230 | 0.0031  (492) |
| 115 | 0.0082  (570) | 235 | 0.0439  (417) |
| 120 | 0.0026  (620) | 240 | -0.0033  (435) |

Average Pearson Correlation Coefficients Between S&P 500 and Keyword Polarity (Own classifier)

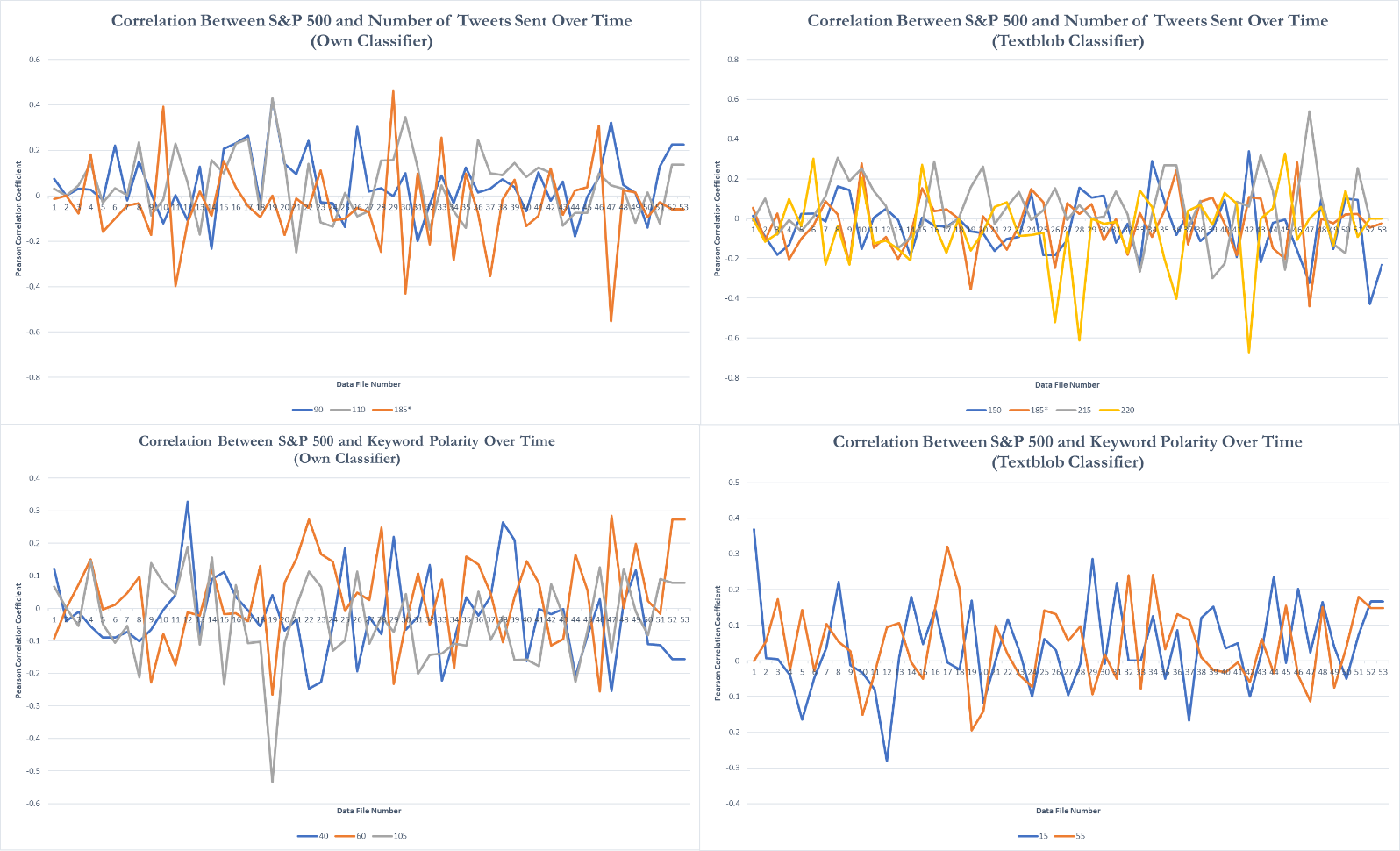
Table 4. Average Pearson Correlation coefficients between the S&P500 returns and keyword polarity, at a given lag in minutes. Uses the sentiment classification algorithm from the Textblob module for Python. Wilcoxon’s W-statistic in parentheses. \*\*\* p < .01; \*\* p < .05; \* p < .10

Table 3. Average Pearson Correlation coefficients between the S&P500 returns and keyword polarity, at a given lag in minutes. Uses the sentiment classification algorithm as trained by the author of the paper. Wilcoxon’s W-statistic in parentheses. \*\*\* p < .01; \*\* p < .05; \* p < .10

Average Pearson Correlation Coefficients Between S&P 500 and Keyword Polarity (Textblob classifier)

Deducing from these tables it is safe to say that the independent variables as measured here have little to no significant effect on the returns of the S&P500. Moreover, it can be seen that the significance of correlations does not hold up across different classifiers. This might indicate that variance, due to the relatively small sample size (N=53), plays a big role in determining the statistical significance of the correlation coefficients being different from zero. There is only one lag, that of 185 minutes, which has weak statistical significance over both classifiers, when only looking at correlations in the same direction between the S&P500 and the number of tweets sent. In favor of the argument that the significance is mainly volatility based, it must be stressed that the sample size is relatively small. When dealing with quantitative finance and predicting stock returns, sample sizes to base conclusions on are usually substantially larger (see i.e., Allen & Karjalainen, 1999; Fernández-Rodríguez, González-Martel & Sosvilla-Rivero, 1999).

However, while looking more closely at the tables as presented, some statistically significant lags (note that no lag is significant at p < 0.01) can be seen. This can nevertheless indicate that some links exist, albeit weak. Since the data as presented above leads to believe that some weak links might exist between keyword polarity and the S&P500 returns, this is further investigated by taking a look at how the correlation coefficients change over time. If it seems to be the case that the correlation coefficient follows the same trend for a long time, but only sporadically dips above or below the trend, there might still be a possibility for a trading strategy. This will however not be reflected in the average correlation coefficient in this case, since it takes the entire sample size into account, treating each observation as equally important, regardless of past and future observations. Moreover, important systemic events (i.e. political issues) might induce bias in the data, which will not be shown when merely taking a look at the average correlation coefficient. Hence, it is shown how the correlation measures change over time for each for some of the lags. Lags that are statistically significant at the 5% level are included in the analysis, in order to ensure for a better representation of the data. The results of this analysis are as follows.



2

4

3

1

Figure 6. Pearson correlation coefficients over time for statistically significant lags. Both classifiers are included, as well as both independent variables. Systemic events that happened during the recording period are marked on the top-right graph. (1) Trump’s tax cut announcement aftermath. (2) Trump’s healthcare bill fails to pass house. (3) First and second round of the French elections. (4) Trump’s budget plan announcement. \* Weakly statistically significant across classifiers

From looking at these graphs it is safe to say that the correlation coefficients vary quite significantly over time. While all of the lags that are included in the visualization are statistically significant at the 5% level, they tend to not follow a similar trend. Furthermore, important systemic events do seem to have some minor effects on the overall trend of the correlation coefficients. This is especially derivable from the bottom-left graph in figure 4, where at times the correlation coefficient differs greatly from its trend, which can be accredited to these important events. However, on other days where the correlation coefficients differ from their trend this does not seem to be the result of a systemic event. An example of this can be seen by looking at file number 19 (the 23rd of March this year), where in most graphs the correlation coefficient greatly differs from its trend, while this does not seem to be the result of any systemic event.

As such, it can be deducted from these graphs that the correlation coefficient tends to follow a random walk over the long-run, even though they are statistically different from zero using Wilcoxon’s signed-rank tests. In addition to this, it may be noted that the implicit assumptions of positive covariance between keyword polarity and S&P500 returns made previously in this paper does not seem to hold. This can either indicate that this paper’s proxy for market sentiment is not truly representative of actual market confidence, or that the expected outcome simply does not hold up in the empirics. Using this knowledge, it would seem fairly difficult to impose a profitable trading strategy on the data presented as such.

1. Discussion

The results of this paper portray that little significant effects can be found between this paper’s proxy of market confidence and the S&P500 index returns. While this paper lays the foundation for further analysis into the topic, it is not thoroughly and convincingly substantiated due to some minor flaws in the method design. Further research into the topic should take note of these flaws, and try to alleviate these issues.

When analyzing the correlation, lagged correlations were established in order to try and predict future returns. In this paper, it was checked whether a lagged change in polarity or the number of tweets sent affected future S&P500 returns. Within the paper, lagged variables were added for each 5-minute time interval, between 5 minutes and 240 minutes. However, adjusting the time interval in which data was stored (e.g. 10-minute time intervals) could affect significance of the results. Therefore, it could be analyzed how different time-intervals, and thus different lags, would affect the results presented by this paper. Moreover, increasing the maximum amount of lag to include daily lags could also yield valuable insights. This would entail analyzing how changes in day *k-y* would affect changes in day *k*, where *y* is the lag in days.

Additionally, there is possibility for access to a bigger dataset. This would assure that the results as presented in this paper would be more thoroughly substantiated, and hence gain in relevance. Gaining access to a bigger dataset could be achieved by collecting data for a longer period of time, or by accessing a historical tweet archive such as provided by GNIP[[5]](#footnote-5). The latter is an authorized reseller of historical tweet data, and would allow for a sample size much greater than what is used in this paper. While it would have been optimal for use in this paper as well, the monetary funds that this requires were not compatible with this author’s budget. Nevertheless, a significant increase in sample-size would lead to more conclusive results.

In this work only one keyword was analyzed, namely “Trump”. While results concerning the overall sentiment surrounding Trump could provide us with interesting insights, it should be noted that a plethora of other keywords are possible. When applying other keywords than that of Trump, or even including other keywords into the analysis, this could provide academia with additional evidence in favor or against the statements laid about by this paper. Additional countries apart from the U.S. could for example be included, by simply adding in a separate keyword which represents the leader of that respective country.

This study has resolved around finding a significant correlation between keyword polarity and the S&P500 index returns. Since some minor statistically significant links have been found in this paper, it would seem of interest to impose a trading strategy based on these links. While the volatility of the correlation found in this paper is quite substantial, applying what was found to be significant could nevertheless lead to additional insights. The chances of such a strategy being highly profitable seem slim, but the academic relevance of attempting such an action seems pertinent. While such a strategy could be profitable, it does not guarantee that causality exists between keyword polarity and the S&P500. Academics should take note of the fact that reverse causality issues and omitted variable bias could influence the results of such a study substantially.

Lastly, adopting a different classification of sentiment regarding tweets could lead to different results. As portrayed by Bollen, Mao and Zeng (2011), human sentiment is multidimensional. The approach to classifying sentiment in this paper might therefore be lacking, as it assumes trinary distinctions in mood. Using the multidimensional model as proposed by the authors of said paper, tracking mood would consist of six states, namely: Calm, Alert, Sure, Vital, Kind and Happy. As implementation of additional variables can usually lead to more predictive models, it would therefore seem of great interest for future research to include this in their works.

1. Conclusion

In this paper, it was investigated whether and how keyword polarity covaries with the Standard and Poor’s 500 index returns. The paper builds on previous research as it attempts to analyze whether general keywords can be taken as a proxy for market sentiment and confidence, and thus affect index returns. The keyword that was used concerned the 45th president of the United States of America: Donald J. Trump. The Twitter Streaming API was used to stream in over 2,500,000 tweets, in the span of 53 trading days, containing in one way or another the keyword “Trump”. Sentiment was extracted from these tweets using two separate machine-learning algorithms, in the shape of Naïve Bayes Classifiers. One of these classifiers was provided by the Textblob module for Python, while the other one was trained by the author of this paper. The difference between these two algorithms was the training data. Where the Textblob module was trained on IMDB movie reviews, the other classifier was fed tweets to train on. While the Textblob module surprisingly yielded higher accuracy scores for extracting sentiment, the classifier that was trained on tweets was also included in the research in order to test for consistency of the results.

Additionally, this paper defined a simple measure of polarity, which was simply taken by dividing the number of positive tweets by the number of negative tweets in a given time span. It was then analyzed whether this measure positively or negatively covaried with the S&P500 index returns. Pearson correlation coefficients were calculated in order to test for this covariance. The results showed no clear relationship between this measure of polarity and the index returns. Wilcoxon’s signed-rank tests were employed to test for statistical significance of these results, since it was expected that there is no normal distribution present around the sample mean. Conclusive evidence was however not drawn from these results, which can be attributed to the small sample size. In addition, it was investigated whether the number of tweets sent surrounding Trump have an effect on the returns of the S&P500. No significant results have been discovered from this analysis either.

Further research into the topic should focus on investigating whether adding different keywords, lags, data or another classification of sentiment could more thoroughly substantiate the claims laid about by this paper.

Reference List

Allen, F., & Karjalainen, R. (1999). Using genetic algorithms to find technical trading rules. *Journal of Financial Economics*, 51(2), 245-271.

Asur, S., & Huberman, B. A. (2010). Predicting the future with social media. In: *2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT),* 1, 492-499.

Bartov, E., Faurel, L., & Mohanram, P. S. (2016). Can Twitter Help Predict Firm-Level Earnings and Stock Returns?. *Rotman School of Management Working Paper No. 2631421.*

Basu, S. (1977). Investment Performance of Common Stocks in Relation to Their Price-Earnings Ratios: A Test of the Efficient Market Hypothesis. *The Journal of Finance,* 32(3), 663-682.

Bollen, J., Mao, H., & Zeng, X. (2011). Twitter mood predicts the stock market. *Journal of Computational Science, 2(1), 1-8.*

Cao, Q., Duan, W., Yu, Y. (2013). The impact of social and conventional media on firm equity value: a sentiment analysis approach. *Decision Support Systems,* 55(4), 919-926.

Fama, E. F. (1965). The behavior of stock-market prices. *The Journal of Business*, *38*(1), 34-105.

Fama, E. F., & French, K. R. (1992). The cross‐section of expected stock returns. *The Journal of Finance*, *47*(2), 427-465.

Fernández-Rodríguez, F., González-Martel, C., & Sosvilla-Rivero, S. (2000). On the profitability of technical trading rules based on artificial neural networks: Evidence from the Madrid stock market. *Economics letters*, *69*(1), 89-94.

Gantz, J., & Reinsel, D. (2012). The digital universe in 2020: Big data, bigger digital shadows, and biggest growth in the far east. *IDC iView: IDC Analyze the future*, *2007(2012), 1-16.*

Jensen, M. C. (1978). Some Anomalous Evidence Regarding Market Efficiency. *Journal of Financial Economics,* 6(2-3), 95-101.

Lo, A., & MacKinlay, C. A. (1999). A Non-Random Walk Down Wall Street. *Princeton:* *Princeton University Press*.

Matsuo, Y., Okazaki, M., & Sakaki, T. (2010). Earthquake shakes Twitter users: real-time event detection by social sensors. *Proceedings of the 19th international conference on World Wide Web,* pp. 851-860.

Pak, A., & Paroubek, P. (2010). Twitter as a corpus for Sentiment Analysis and Opinion Mining. *LREc, 10*(2010).

Samuelson, P. A. (1965). Proof That Properly Anticipated Prices Fluctuate Randomly. *International Management Review,* 6, 41-49.

Sandner, P. G., Sprenger, T. O., Tumasjan, A., & Welpe, I. M. (2010). Predicting elections with twitter: What 140 characters reveal about political sentiment. *ICWSM*, *10*(1), 178-185.

Wilcoxon, F. (1945). Individual comparisons by ranking methods. *Biometrics bulletin,* 1(6), 80-83.

Appendix

In order to check for the expectation that the correlations are non-normally distributed across lags, frequency tables have been made to visually portray this expectation. Correlations have been rounded to the nearest 0.05 digit as this allows for the frequency distribution. While it would be more representative to not round the correlations, creating these graphs would be impossible due to the granularity of the data. The first and the last lag have randomly been selected for visualization. It can be derived from the graphs that the data does not seem to be normally distributed.

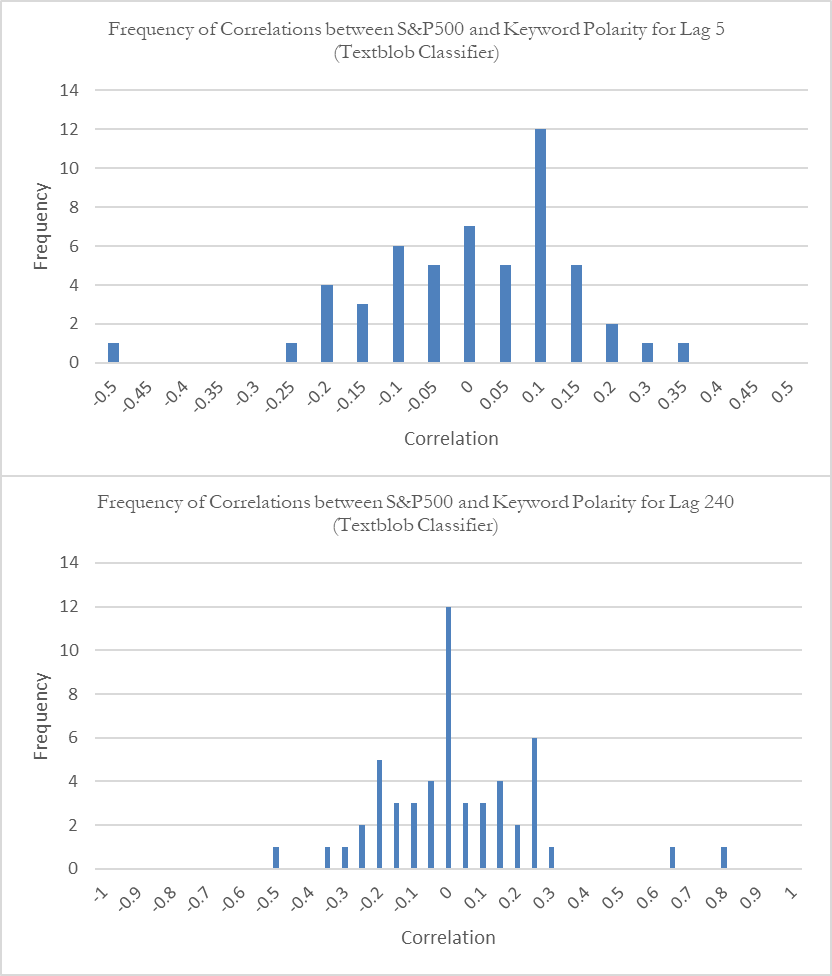


Figure 7. Frequency distribution of Pearson correlation coefficients for the first and last lag. Only correlations between S&P500 and keyword polarity are taken into account.

1. https://dev.twitter.com/streaming/overview [↑](#footnote-ref-1)
2. https://textblob.readthedocs.io/en/dev/ [↑](#footnote-ref-2)
3. A probabilistic classifier that is based on Bayes’ theorem with naïve independence assumptions between features. For more information: http://www.ic.unicamp.br/~rocha/teaching/2011s1/mc906/aulas/naive-bayes.pdf [↑](#footnote-ref-3)
4. http://www.nltk.org/api/nltk.corpus.reader.html#module-nltk.corpus.reader.twitter [↑](#footnote-ref-4)
5. https://gnip.com/ [↑](#footnote-ref-5)