CHARLES UNIVERSITY

FACULTY OF SOCIAL SCIENCES

Institute of Economic Studies

# Bachelor thesis

2017 **Jan Hynek**

CHARLES UNIVERSITY

FACULTY OF SOCIAL SCIENCES

Institute of Economic Studies

**Jan Hynek**

# Stock market prediction using Twitter

## Are we able to identify the market trends from specific words on Twitter?

*Bachelor thesis*

Prague 2017

**Author**: Jan Hynek

**Supervisor**: doc. PhDr. Ladislav Krištoufek Ph.D.

**Academic Year**: 2016/2017

## Bibliographic note

HYNEK, Jan. *Stock market predicition using Twitter. Are we able to identify the market trends from specific words on Twitter?* Prague 2017. 35 pp. Bachelor thesis (Bc.) Charles University, Faculty of Social Sciences, Institute of Economic Studies. Thesis supervisor doc. PhDr. Ladislav Krištoufek Ph.D.

## Abstract

In this work I examine the short-time relationship of Twitter on the markets. I had been downloading English tweets in the period between 9th March and 4th April and also tweets containing words and hashtags "apple", "microsoft", "boeing", "cocacola". Afterwards, I investigate the predictive power of frequency of individal words on the marke using multinomial and binomial penalised logistic regression. I conclude that this method cannot be used for prediction, but can provide interesting insight ex-post.

## Abstrakt

V této práci jsem se zabýval krátkodobým vlivem Twitteru na trhy. Stahoval jsem anglické tweety z období mezi 9. březnem a 4. dubnem, společně s tweety obsahující slova a hashtagy "apple", "microsoft", "boeing", "cocacola". Následně jsem zkoumal pomocí multinomiální a binomiální penalizované logistické regrese, jestli je možné predikovat trhy pomocí frekvence slov na trhu. Po použití out-of-sample predikce jsem zjistil, že tato metoda není vhodná pro predikci trhů, ale může poskytnout ex-post zajímavý vhled do vztahu sociálních sítí a trhů.

## Keywords

Twitter, document-term matrix, logistic regression, text mining, sentiment analysis, market prediction

## Klíčová slova

Twitter, document-term matrix, logistická regrese, text mining, sentiment analysis, predikce trhu

## Declaration of Authorship

I hereby proclaim that I wrote my bachelor thesis on my own under the leadership of my supervisor and that the references include all resources and literature I have used.

I grant a permission to reproduce and to distribute copies of this thesis document in whole or in part.

Prague, 17 May 2017

Signature

## Acknowledgment

I would like to express my gratitude to doc. PhDr. Ladislav Krištoufek, Ph.D. who provided me with interesting ideas and helped me to shape this thesis.
Last, but not least, I would like to thank my fiancée Kamila, who was supporting me throughout the time of my studies.

## Bachelor Thesis Proposal

| | |
|---|---|
| **Author** | Jan Hynek |
| **Supervisor** | doc. PhDr. Ladislav Krištoufek, Ph.D. |
| **Proposed topic** | Market prediction using Twitter sentiment analysis |

### Research question and motivation

Is it possible to predict market movements using Twitter sentiment? Can we improve sentiment analysis using word2vec algorithm?

The influence of the social media on the current society is one of the biggest game-changers in trading. Even though efficient market hypothesis states that "stock market prices are largely driven by new information and follow a random walk pattern", it seems evident that a single tweet can cause to move a specific stock go up or down. Such can be the case the Twitter itself – when their disappointing results leaked online, its stocks decreased by 20%. And as the whole Twitter is composed from information bits like this, it seems intuitive that if we analyse all of the tweets it might be possible to predict the movement of the whole market.

### Methodology

We are going to use publicly available SNAP Twitter dataset consisting of 476 million tweets from period between June to December 2009. We will analyse every tweet and and assign him value of its overall sentiment. The widely spread method in the sentiment analysis research is lexicon based sentiment analysis. The main essence of this method is lexicon of emotionally tinged words, and an algorithm looking at every sentence in a given corpus. Even though this method is generally effective, it has its drawbacks. If we consider word such as 'death', it is generally considered that it reflects negative sentiment. But in the case of one-time events such as death of Osama bin Laden that might not be true and therefore our results could be biased on such days. I would like to use small lexicon of so called "anchor words" – words unambiguously expressing certain emotion only. We will obtain the words from Profile of Mood States (POMS) questionnaire. Then, using word2vec algorithm, we will obtain similar words to every given anchor word for every day. Using these, I will obtain the sentiment of the given tweet on given day. I will sum the obtained sentiment. In the end, I will regress DJIA on the obtained sentiment from the previous day.

### Contribution

Even though there is a lot of companies focused on sentiment trading, there has been done only little research. We can evaluate whether these methods are capable of predicting market movements. Other contribution is the new method of sentiment analysis. This method should theoretically be robust to unexpected events and therefore this method could have more precise results.

### Outline

1. Introduction
2. Literature review
3. Explaining word2vec algorithm
4. Exploratory dataset analysis
5. Sentiment analysis
6. Model learning
7. Prediction of Dow Jones Industrial Average
8. Discussion of results
9. Further work
10. Literature

**List of academic literature**

- Bollen, Johan, and Huina Mao. 2011. “Twitter Mood As A Stock Market Predictor”. *Computer* 44 (10): 91-94. doi:10.1109/MC.2011.323.

- Rexha, Andi, Mark Kröll, Mauro Dragoni, and Roman Kern. “Polarity Classification For Target Phrases In Tweets: A Word2Vec Approach”, 217. doi:10.1007/978-3-319-47602-5_40.

- MARTÍNEZ-CÁMARA, EUGENIO, M. TERESA MARTÍN-VALDIVIA, L. ALFONSO UREÑA-LÓPEZ, and A RTURO MONTEJO-RÁEZ. 2014. “Sentiment Analysis In Twitter”. *Natural Language Engineering* 20 (01): 1-28. doi:10.1017/S1351324912000332.

- Makrehchi, Masoud, Sameena Shah, and Wenhui Liao. 2013. “Stock Prediction Using Event-Based Sentiment Analysis”. *2013 Ieee/wic/acm International Joint Conferences On Web Intelligence (Wi) And Intelligent Agent Technologies (Iat).* IEEE, 337-342. doi:10.1109/WI-IAT.2013.48.

- Zhang, Xue, Hauke Fuehres, and Peter A. Gloor. 2011. “Predicting Stock Market Indicators Through Twitter “I Hope It Is Not As Bad As I Fear””. *Procedia - Social And Behavioral Sciences* 26: 55-62. doi:10.1016/j.sbspro.2011.10.562.

# Contents

**Introduction** **1**

**1 Literature Review** **5**

**2 Methodology** **8**

2.1 Procedure overview . . . . . . . . . . . . . . . . . . . . . 8

2.2 Datasets . . . . . . . . . . . . . . . . . . . . . . . . . . . 9

2.3 Text classification, logistic regression and prediction . . . . . 14

**3 Results** **19**

3.1 Document - term matrices . . . . . . . . . . . . . . . . . . . 19

3.2 Small datasets . . . . . . . . . . . . . . . . . . . . . . . . . 20

3.2.1 Generalisation performance of the multinomial model 21

3.2.2 Binomial models and their generalisation performance 21

3.2.3 Investigating the case of Boeing and finding market relevant words . . . . . . . . . . . . . . . . . . . . . . . . 23

3.3 "All English tweets" dataset and problems of Big Data analysis 27

**Conclusion** **29**

**List of tables and figures** **35**

**Appendix** **37**

# Introduction

Social media has engaged hundreds of millions of users nowadays. Majority of them are actively creating data about themselves and their preferences toward various subjects, such as political issues or company products. And that sparked interest of marketing analytics, politicians and researchers.

Main problem is that the data on social media are created at a fast rate. If we consider Facebook, its amount of data stored on average day in 2014 summed up to 600TB. However, developers also state that this amount tripled from the previous year, even though the amount of users rose only by 18%[1][2]. This rate of new data creation, along with its variety and its overall abundance, is commonly referred to as *Big Data*[3]. Even though a important part of these data is publicly accessible, extracting useful information is a challenge. In the case of online marketing, stored data do not provide value for social network providers or advertisers, until they can find relevant information about customers and their interests and successfully target them.

Similar applies for researchers. Even though many of them are aware of the easy access to vast amounts of data available on social networks, conducting analysis which is meaningful, contributive and insightful is demanding task, yet many researchers claim that they achieved this. In this thesis, I will look deeply at the contribution connected with stock market prediction, where several researchers claim they have achieved valuable insight, such as in widely cited research made by Bollen, Mao and Zeng (2011).

Firstly, current research needs to evaluate used methods. To use words in market prediction, the most common method usually is *Sentiment Analysis.* This is an umbrella title for computational methods for opinion extraction from given textual data and its classification, typically whether the extracted opinion is positive, negative or neutral[4]. However, most research share the

[1] code.facebook.com, Scaling the Facebook data warehouse to 300 PB: https://goo.gl/2OeUOE

[2] statista.com, Number of monthly active Facebook users: https://goo.gl/k4DdxG

[3] This term does not apply at the social networks only. It can be used in any industry.

[4] $https://en.oxforddictionaries.com/definition/sentiment\_analysis$

same pattern:

1. Obtain raw text
2. Apply text mining to simplify data
3. Apply sentiment analysis
4. Use obtained sentiment for stock market prediction

The problem I am investigating in this thesis is that the sentiment analysis mostly uses arbitrarily chosen words, which usually have positive or negative connotation. However, there were no research done whether these words really have effect on markets. Therefore, I decided to simplify this task:

1. Obtain raw text
2. Apply text mining to simplify data
3. Use simplified text data for stock market prediction

For stock market prediction, I use logistic model where I am able to investigate the coefficients on the individual words. Therefore, I can evaluate which words have the biggest influence on the stock market.

Secondly, most of the research aimed at stock market prediction using Twitter data has been aggregated by days and I find this granularity insufficient nowadays. We can observe that in many cases markets react instantly[5]. To illustrate this point I want to present the case of the United Airlines. On 11th of April 2017, plane crew kicked out a passenger out of the fully booked plane, and this event caused huge outrage, especially on social media. This event happened during inter-day period of the stock market, and caused drop in the United Airlines share price by 4% when markets opened, however by the end of the day, the price almost recovered. At this moment, we cannot distinguish this event while looking at the daily stock price. Therefore, I assume that if we took the daily close price of United Airlines and as well as all tweets posted on Twitter in the previous day, we would not

[5] http://www.cnbc.com/2016/12/06/can-algos-trade-trumps-tweets-absolutely-maybe.html

see any significant relationship. Therefore, I chose the half hour granularity while the shares are traded and I am treating the inter-day periods as a single period. This approach is innovative in research, and this allows us to examine short-temporal relationship between social media and markets.

Thirdly, the theoretical background in this field is not strong enough and definitely needs further work in the future. It is important to answer why it could be possible to predict markets through social media. Recently, Moat, Olivola et al. (2016) suggest that with the advancement of internet, we can observe decision making process of huge amount of individuals. Authors suggest that if we aggregate human decisions observed via internet searches performed or reviews and posts written, it might be possible to link this to real life examples of decision making, such as market movements. This is an assumption on which most of the research, even though not stated, relies. However, the main scope of this thesis is not to deepen the theoretical background but rather to evaluate commonly used methods and also to present new approach to temporal granularity. Still, I believe it is important to deepen the theoretical framework in this field and this work could serve as a basis for a new theoretical framework. Basic notions can be noticed in the example of the United Airlines.

Finally, using social media sentiment to predict the stock market is however in direct contradiction with Efficient Market Hypothesis by Fama (1965) and Fama (1970). One of the most widely cited definition is by Malkiel (1989), formualted as follows:

> A capital market is said to be efficient if it fully and correctly reflects all relevant information in determining security prices. Formally, the market is said to be efficient with respect to some information set, $\Omega_t$, if security prices would be unaffected by revealing that information to all participants. Moreover, efficiency with respect to an information set, $\Omega_t$, implies that it is impossible to make economic profits by trading on the basis of $\Omega_t$.

Timmermann and Granger (2004) revised this definition incorporating search methods and models which he consider significant as well. He states that markets are overall efficient, however he claims that there exist a window when there are profits possible between introducing a new search method or model until widespread adaptation by majority of market participants. It is also worth noting that Malkiel (2003) also admits that a market cannot be perfectly efficient, however states that any irrationalities or patterns in the pricing won't stay there for long. And even though irregularities do not stay long on the market, Malkiel states that this motivates researchers to look for them. And this is the main motivation of this thesis, to find a new searching method for market irregularity. Therefore, I am asking whether we are able to identify the market trends from frequency of specific words on the Twitter. In order to evaluate this question, I perform out-of-sample prediction on the data. This will allow me to evaluate the research question as well as validity of the words found in the model.

In the next section, I will review related literature. This will be followed by description of methodology and datasets and I will analyse the results afterwards and compare them with research of other authors and I will conclude with discussion about contribution of this thesis.

# 1 Literature Review

In the following section, I first look at the stock market prediction using Twitter. Afterwards I look at application of Twitter data for predicting elections or box-movie revenues. In the end I look at the criticism of these methods, reaction at the criticism and also look into some criticism - compliant methods.

Bollen, Mao and Zeng (2011) were one of the first to claim they found market irregularity on social media. They used Twitter to obtain tweets from March 2008 to November 2008. Firstly, they filtered all tweets containing expressions "I feel" or "I am". Secondly, they obtained sentiment and classified each tweet with two methods: OpinionFinder used Positive/Negative classification, GPOMS Mood classification consisted of 6 mood dimensions (Calm, Alert, Sure, Vital, Kind, and Happy). They found significant relationship via Granger causality analysis of Dow Jones Industrial Average (DJIA) with the GPOMS Calm dimension lagged by 2 to 6 days. They also used Self Organising Fuzzy Neural Networks (SOFNN) to predict market values and were able to predict 13 out of 15 days with GPOMS Calm dimension, claiming 86.7% accuraccy. These results were verified by Mittal and Goel (2012) where they applied another methods e.g. Support Vector Machines and concluded that using Twitter sentiment for stock market prediction is possible. However, Bollen, Mao and Zeng (2011) do not provide any intuition why Calmness of the general public should predict Dow Jones index, only concludes that the results are *'strongly indicative of a predictive correlation'*, however the team is aware of this and state that finding the link *'remains a crucial area for future research'*.

The prediction of voting results is also a common theme for research and several studies has been conducted. Most of them had concluded that it is possible to use sentiment analysis on Twitter for forecasting elections (Tumasjan et al. 2011). However Gayo-Avello (2013) conducted meta-analysis of Twitter electoral predictions and stated that the biggest weakness of all analysis is that all of them are conducted post-hoc. He also stated that the sentiment analysis is usually conducted with naïveté and therefore ignoring humour and sarcasm, and that these analysis usually ignore the problem of population bias as well. Other than that, Asur and Huberman (2010) used count of messages on Twitter containing movie names and predicted the box office movie revenues. They had shown that in combination with sentiment analysis this predictor can outperform predictors using artifical Hollywood Stock Exchange. There were also other approaches to predict decision making processes. Preis, Moat and Stanley (2013) used Google Trends for 98 financially oriented terms and found significant link between most of the searched words and DJIA. They concluded that most important ones for the market prediction are "color" and "debt". Other than that, Moat, Curme et al. (2013) examined the relationship between markets and Wikipedia edits. They had found increased number of edits in articles connected with financial markets in the period before financial crisis, and at the same time they have not observed similar behaviour in the articles connected with movies.

### Criticism and reaction

Yet Lazer et al. (2014) illustrate the traps of the Big Data analysis on the example of the Google Flu, an algorithm using Google Trends to predict influenza epidemics. In this article authors point out that even though algorithm used to be precise, changes in the search algorithm, primary subject of interest for Google, biased the results of the Google Flu. Authors also criticise that Big Data methods are often very distinct from traditional statistical methods and ignore problems connected with statistical assumptions and argue that their result were often worse. They illustrate this point on the example of regressing current estimates made by Centers for Disease Control and Prevention (CDC) on the past one and comparing these results with Google Flu Trends. However, they also suggest that the added value of the Big Data analysis is in understanding *'the prevalence of flu at very local levels, which is not practical for the CDC to widely produce'*.

This article triggered the reaction of Preis and Moat (2014) who suggest that using Google Flu Trends might provide insightful data, when used together with historical data in adaptive nowcasting models. This is also motivation of the work by Kristoufek, Moat and Preis (2016), who also used Google Trends data to predict suicide rate in England. They concluded that as the data for suicide rates are available after 2 year lag, Google Trends might provide suitable estimation in the meantime. In line with the Lazer et al. (2014) suggestions, there has been done several granular sentiment analysis research focused on prediction. Eichstaedt et al. (2015) used Twitter sentiment to predict coronary diseases on the county level. Gerber (2014) used Twitter to obtain main topics in specific area of the city and used these data to make crime prediction more precise.

It is common nowadays that markets react immediately to statements sent to Twitter. Namely, tweets of Donald Trump has been subject to trading algorithms, which exploited the fact that when current U.S. president Donald Trump tweeeted, markets reacted rapidly[6][7]. In this thesis I will examine this short-term relationship of markets and social media and find whether there are any universal signs associated with significant market growth or decrease.

[6] http://fortune.com/2017/01/05/stocks-trump-tweets/

[7] http://www.cnbc.com/2016/12/06/can-algos-trade-trumps-tweets-absolutely-maybe.html

# 2 Methodology

In the following section, I will present procedure used throughout the analysis, performed in R. I explain individual steps, as well as what document - term matrix is and which yields were identified as significant in more detail afterwards.

## 2.1 Procedure overview

1. Download all tweets for chosen companies, download latest 5000 English tweets every 5 minutes. Altogether 5 different Twitter datasets.
2. Get corresponding stock prices or Dow Jones Index for every half-hour
3. Identify half-hour periods, when yields were significantly positive or negative and create two sets of dummy variables.
4. At a arbitrarily given time, take downloaded tweets and simplify them to time and tweet text only.
5. Divide each Twitter dataset in two. Training dataset, consisting of first three quarters of observations, and testing dataset consisting of the rest.
6. For each tweet, use the tweet timestamp to attach the variable indicating the significantly positive or negative yields.
7. Create document-term matrices from all tweets. Rows are individual tweets, columns are the words which also are explanatory variables
8. Crossvalidate penalised multinomial logistic regressions to obtain the optimal value of penalisation coefficient.
9. Repeat the previous step, but using binomial logistic regression.
10. Take the optimal multinomial and binomial model and use it to predict values from the testing dataset.
11. Compare the results obtained from prediction with actual values using confusion table.

## 2.2 Datasets

In this thesis I worked with two types of datasets, Twitter datasets and financial datasets. I aimed to connect them together to obtain explained variable, which consists of information about future market movement, when individual tweet was posted. Explanatory variable is created by number of words in this tweet. In the next section present the details.

First, I had to obtain Twitter dataset. My original thought was to use Stanford database of 476 millions of tweets from period between February and December 2009[8]. However based on the work by Lazer et al. (2014) who stated that frequent changes on the websites make a lot of research done on older data irrelevant for these days, I realised that this is also the case of the Twitter. Twitter allowed to add photos, and also allowed for longer tweets. Therefore, I looked for more recent dataset, however I have not found any as Twitter terms of service forbids to share such datasets publicly. Therefore, I decided to download my own dataset. I downloaded latest 5000 English tweets every five minutes between 9th March 2017 and 4th April 2017, and I will call this dataset onwards as an *"English tweets dataset"*. I downloaded also all available tweets containing following words: *microsoft, #microsoft, apple, #apple, boeing, #boeing, cocacola, #cocacola* between 19th March 2017 and 4th April 2017. I created 4 individual datasets for every individual company, and I will call them onwards as *"Company tweets datasets"*. Their collection required 24 / 7 running server as Twitter API [9] do not allow to download more than either 5000 tweets from the past, or at most 7 days. Still, I had to manage several technical problems, which resulted in paused download. Therefore, we can observe several dents in the data, which can be observed in figures 1 and 2. I decided to divide the datasets in two parts, one with approximately $\frac{3}{4}$ of observations, which will be used for training the models. The rest is used for testing predictive performance of the models. This can be also seen in figures 1 and 2.

Second, it is important to say that in this format of data, I would be

[8] Available here: https://snap.stanford.edu/data/twitter7.html

[9] Application Programming Interface

unable to work with such dataset, therefore, I decided to clean the text and build *Document-Term Matrix* (DTM). To explain what DTM is, I will discuss first how DTM is created. This method relies on the strong assumption, that the only the words itself carry the information, and the order of the individual words in the sentence, paragraphs and other things does not matter. This approach is called *Bag of Words* modelling. Now we can simplify the work with words and quantify them. We can assume that every individual tweet can be considered as a small bag of words and count individual number of words inside.

To further explain, consider following example of tweet made by Donald Trump on 6th December 2016. It caused immediate drop of Boeing shares.

> *Boeing is building a brand new 747 Air Force One for future presidents, but costs are out of control, more than $4 billion. Cancel order!*

This individual tweet is first preprocessed. I decided to make all letters lowercase and omit all symbols, however decided to leave the numbers, as we can see that 747 can be significant. I also decided to omit stopwords (words with low semantical value). The tweet becames:

> *boeing building brand new 747 air force one future presidents costs out control more than 4 billion cancel order*

We can observe that this sentence still have some meaning, however now we could interpret it in several ways. Now, we are able create document-term matrix for this individual tweet. Illustration can be seen in in table 1.

| **Words:** | 4 | 747 | air | boeing | billion | brand | building | costs | ... |
|---|---|---|---|---|---|---|---|---|---|
| **# in tweet:** | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | ... |

Table 1: Illustration of document term matrix for single Donald Trump's tweet.

It is evident that the meaning of the original sentence is dissolved in this matrix and if we would see this table, we would not be able to reconstruct the original tweet or even get the original thought. But for simplicity, I assume that this does not matter and that information in the words alone is sufficient for prediction.

In the end, I used *text2vec* package to do this process for every single tweet in the given dataset. Now every single row is considered as a single tweet, and every column indicates a single column. Number, which coordinates are $[x, y]$ indicates number of words $y$ in a tweet $x$. Illustration of such table can be seen in table 2. But we can observe, that such tables are very sparse - most of the tables are zero and only occasionally do have some value. This is logical, as a single tweet can contain at most 30 short words, but to cover at least 72% of commonly used dictionary, we need at least 1000 words (Nation and Waring 1997). Therefore, to have meaningful compromise between the richness of the vocabulary and performance of the algorithm, I decided to prune the words if they were in less than 0.01% of tweets or had less than 10 occurrences, and this resulted in DTMs with around 1500 distinct words. Another positive side-effect also was that this reduced a lot of sparsity in the data. This concludes creation of explaining variables in this thesis.

| **Words:** | apple | art | boeing | building | civil | fake | french | costs | ... |
|---:|---:|---:|---:|---:|---:|---:|---:|---:|---|
| **Trump's tweet** | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | ... |
| **rand. tweet 1** | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | ... |
| **rand. tweet 2** | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | ... |
| **...** | | | ... | | | ... | | | ... |

Table 2: Illustration of pruned document term matrix. In this illustration we can observe that several words were omitted from the Donald Trump's tweet.

In order to obtain explained variables, I had to download financial dataset. I decided to use Google Finance API. It allows to download data with prespecified periodicity up to ticks, and I decided to use half-hour periodicity. I downloaded Dow Jones Industrial Average (DJIA) index data, which I used to create explained variable for the "All tweets" dataset. I downloaded dataset, which started one day before the first tweet from "All tweets" and ended one day after the last one to ensure overlap. Afterwards, I obtained yields by subtracting Close value at time $t-1$ from Close value at time $t$. Then, I created dummy variable signalling when these yields were at least one standard error above the mean and another set signalling when it was one standard error below. In order to inspect predictive power of the tweets, I decided to lag these dummy variables one period backwards. I applied exactly same approach also for the information about stocks of the individual companies, and ended up with 5 datasets with financial information. It should also be noted, that I treated inter-day and inter-week periods as a single period, as it seems intuitive that if any event happened during this time, it would be visible in the opening price. The problem here is that in the given half hour I collected up to 30000 tweets. I decided to assign them the same value of yields as was in the given half - hour. Therefore, once I had both these datasets, I subtracted the dummy variable indicating low yields from the dummy variable indicating high yields, I assigned its values indicating the yields to every tweet and, and created explained variable for multinomial logistic regression. However, this presented framework also works for downloading single dummy variable so I was also able create dummy variable indicating negative yields and run binomial logistic regression afterwards.

When I had my data prepared, it was time to run analysis. In the following section, I will explain how does text classification work, why is logistic regression used, and how it is implemented in glmnet package.

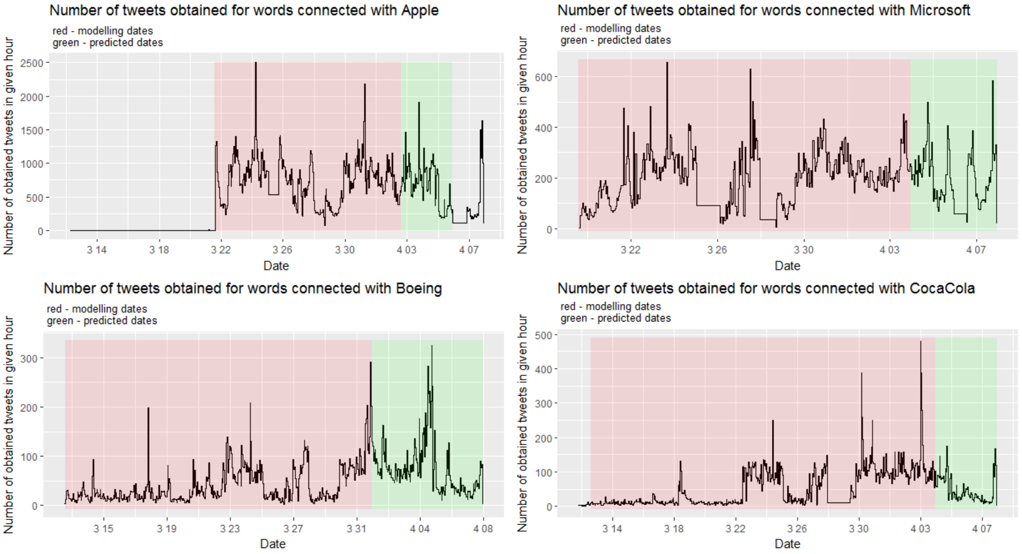


Figure 1: Number of obtained words for individual companies. Red parts are the parts used for modelling the data, green parts are showing when the datasets were tested.

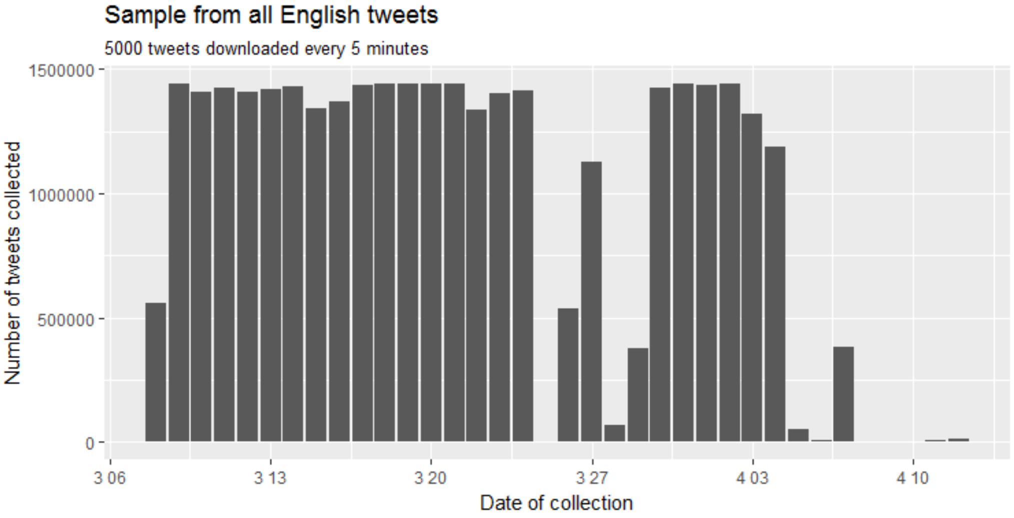


Figure 2: Number of obtained words for sample from all tweets

## 2.3 Text classification, logistic regression and prediction

The basic approach to text classification is that we are trying to assess $n$ classess to any number of documents. In the simplest case we take $n = 2$. Most of the methods used break the document into single words, just as was done in the document - term matrix and assign some values to its words, which is often refered to as *sentiment.* We can aggregate this value afterwards and say that this document has positive or negative sentiment. And if we have pre-classified documents (e.g. manually), we can learn *some model* to classify the documents manually. This method is often used for review classification, where company needs to know the overall satisfaction with its product, and it can provide good results. I modified this method and decided to bypass the sentiment analysis. I am examining the direct effect of tweets at the markets, and therefore I am classifying the tweets naïvely by classifier, which says that the markets significantly moved in the next half-hour. This is also the classifier I am predicting in the out-of-sample analysis.

In the previous paragraph I mentioned that we might use *some model* for automatic text classification. There is a lot of distinct models used in the text classification problems. To name few, Bollen, Mao and Zeng (2011) used neural networks, Mittal and Goel (2012) used support vector machines. In *Speech and Language Processing* by Jurafsky and Martin (2014), authors suggest using Naïve Bayes method. However I decided to use logistic regression, which is considered as a golden standard for text classification. First, it does not work as a black-box model, therefore I am able to extract coefficients for individual words and interpret them (even though interpretation is tricky). Second, it is good choice for highly dimensional and sparse data, such as in this case. Its results can be also made even more accurate (even though slightly biased) by introducing penalisation term (Friedman, Hastie and Tibshirani 2009). Last but not least, I am familiar with basic logistic regression from Econometrics classes in our school.

To shortly explain what how logistic works, I have decided to explain the binomial case only, since it simplifies the explanation and does not differ from the multinomial case a lot. The need for binomial logistic regression arises from desire to model linear probability function of two classes. There exist linear probability model, however that achieves values outside interval $(0, 1)$ and also suffers from heteroscedasticity. The logistic model solves this problem, and at the same time it also ensures that probabilities sum to one. Let G be the set of classes, in this case $\{0, 1\}$. The binomial model takes following form (Friedman, Hastie and Tibshirani 2009):

$$\log \frac{Pr(G = 0|X = x)}{Pr(G = 1|X = x)} = \beta_0 + \beta_1 x \tag{1}$$

This is called the log-odds or logit transformation of the model. In the multinomial case, if we would have $K$ classes, we would take $K-1$ equations, as we need one base class, in this case we took $G = 1$ as the base. As we are interested in the probability of individual class, we have to calculate individual logistic functions, which attain values between 0 and 1 and also sum up to one. After simple calculations, we arrive to the following result (Friedman, Hastie and Tibshirani 2009):

$$\begin{aligned} Pr(G = 0|X = x) &= \frac{e^{\beta_0 + \beta_1 x}}{1 + e^{\beta_0 + \beta_1 x}} \\ Pr(G = 1|X = x) &= \frac{1}{1 + e^{\beta_0 + \beta_1 x}} \end{aligned} \tag{2}$$

This regression is fitted to observed values via maximum likelihood. We usually take the log likelihood for easier computations, and maximise the conditional likelihood of G given X. In this case of binomial logistic regression, we maximise likelihood function, which can be observed in equation 3 and we set its derivatives to zero. This is the basic intuition how logistic regression is implemented in glmnet package and for further details (i.e. the exact way how algorithm converges to maximal value), see *Elements of Statistical learning* by Friedman, Hastie and Tibshirani (2009) or *Regularization Paths for Generalized Linear Models via Coordinate Descent* by the same authors.

$$\begin{aligned}\mathcal{L}(\beta) &= \sum_{i=1}^{N} \big[ y_i \log Pr(G=0|X=x) + (1-y_i)\log Pr(G=1|X=x) \big] \\ &= \sum_{i=1}^{N} \big[ y_i \beta x_i - \log(1+e^{\beta x_i}) \big]\end{aligned} \tag{3}$$

In this thesis I also decided to work with penalisation term. To explain why, it is important to say the basic intuition behind the *Bias - Variance tradeoff*. But to explain that, we have to discuss OLS first. We know that if all assumptions hold, OLS will be BLUE, or best among all linear unbiased estimators. However, the variance of such estimates still can be high, and in exchange for little bias, it might be possible to get far more accurate results. There is another reason why we should introduce penalisation term, which comes from the premise behind OLS, which is following: Even though variance is unwanted, in the long run, analysts will get rid of it with more and more data, and therefore we have to take care of the bias. That said, usually analysts are given one set of data, and there is no long run (Fortmann-Roe 2012). Therefore, we should account in the models for both bias and variance and that is the main reason to introduce penalisation term. It is important to see, how this parameter is implemented in the glmnet package[10] (Friedman, Hastie and Tibshirani 2010):

$$\min_{(\beta_0,\beta)\in\mathbb{R}^{p+1}} -\left[\frac{1}{N}\sum_{i=1}^{N} y_i \cdot (\beta_0 + x_i^T\beta) - \log(1+e^{(\beta_0+x_i^T\beta)})\right] + \lambda P_\alpha(\beta) \tag{4}$$

where

$$P_\alpha(\beta) = \big[(1-\alpha)||\beta||_2^2/2 + \alpha||\beta||_1\big] \tag{5}$$

$P_\alpha(\beta)$ is the penalisation term. If we set $\alpha = 0$, we speak about *Ridge* regression, which prefers smaller values of the regressors, and therefore lowers the variance. Other commonly used method is *LASSO* regression, which I used. We set $\alpha = 1$, and $P_\alpha(\beta)$ becomes $||\beta||$. Its nature allows that it can be used as an automated *variable selector*. This might be often unwanted in

[10] In the previous text I said that glmnet package maximises loglikelihood, however in reality it minimises negative loglikelihood, however both these methods are the same.

the modelling as we would sometimes omit the important variables, however in this case we can assume that I do not need to control for any word and that all words should be treated equally, and every word can be omitted if without a significant effect. Lasso also allows for *parameter shrinkage.* This is direct implementation of Bias - Variance tradeoff. If the coefficient of the penalization function $\lambda$ would be too high, all parameters would shrunk to zero. This would lead to null variance but at the same moment with huge bias. However, if we would set $\lambda$ to zero, parameters would be unrestricted and unbiased, but also with higher variance (Friedman, Hastie and Tibshirani 2009).

This also leads to the question, how do we set parameter $\lambda$ in the regression. One of the commonly used methods is *crossvalidation.* This method is a way to estimate in-sample expected prediction error (EPE) of the given model. In this thesis I specifically used $k$-fold crossvalidation. This means that for each possible value of $\lambda$, the dataset is divided in $k$ different parts, and afterwards, the penalised logistic regression with is run on $k-1$ of them, and used to predict on the last of them. This is done $k$ times altogether, and in the end each part is $k-1$ times in the training set and exactly once in the testing set. And this process is used for each value of $\lambda$ to calculate EPE of the given model. However, the crossvalidation also takes huge amount of computational resources and it was not possible to crossvalidate the results for the "All English tweets" dataset. Therefore, to choose the model with the lowest amount of EPE, I decided to use corrected Akaike and Bayesian Information Criteria. Both of these methods take likelihood of the model and in order to prevent overfitting, they introduce penalty for number chosen variables. The lowest AICc and BIC indicates the lowest expected prediction error. For further details I strongly recommend Elements of Statistical Learning by Friedman, Hastie and Tibshirani (2009).

To evaluate the generalisation performance of the chosen model, I use *out-of-sample* prediction for which I reserved independent test dataset (part not used for training the model; this division can be seen in figure 1). For each tweet in this dataset, I decided to predict, whether it will have positive, neutral or negative effect on the market. Afterwards I compare them with the actually assessed values and evaluate the results using *confusion table*. This is table of size $n * n$, where $n$ is the number of labels, in my case $n = 3$ (Kohavi and Provost 1998). It contains information about actual and predicted values. From this table, we can easily calculate false and true positive and negative rates and tell, whether the classifier is performing better than benchmark. In our case, the benchmark is to classify all observations with the same class. To evaluate whether classifier is effective, we test the hypothesis that our accuracy (rate of correctly identified classes) is greater than the most frequent class (Friedman, Hastie and Tibshirani 2009).

# 3 Results

In this section, I discuss how the document - term matrices look like, their size and their composition. Afterwards, as the work was different with company datasets and "All English tweets" dataset, I decided to split the following section in two. In the section regarding small datasets, I first look at the results from modelling. I discuss the results from multinomial models and discuss which models are chosen. Second, I look at the results from prediction and evaluate confusion matrices. Third, since the models had slightly better performance in predicting negative class, I decided to run binomial logistic regression and try to predict specifically this class, and I will look at these results as well. Last, from all the results, I identify the most interesting case of the Boeing dataset, therefore I will look more deeply into this one. Next, I look into the "All English tweets" dataset and I discuss the computational difficulties connected with Big Data analysis, perform binomial logistic regression, and evaluate the results of the out-of-sample prediction.

## 3.1 Document - term matrices

From tweets created, we created 5 individual document - term matrices. The lowest number of words had the "All English tweets" dataset. This was handy, especially because less words reduced the need for computational resources dramatically, and this was probably caused by the condition that every word in this table have to be in at least 0.01% of tweets. In the case of the smaller datasets, each DTM had around 1500 hundred words. I think that the restrictive condition in this case rather was that each word has to be in at least 10 tweets. The exact number of dimensions can be seen in table 3. Overall, if we discuss the richness of the vocabulary with Nation and Waring (1997), where authors state that 1000 words is enough to capture around 72 % of the used vocabulary I would expect that I also captured around 75% of the vocabulary used. But I omitted on purpose the most commonly used words, and many of the captured words also are nonsensical URL strings of often shared pictures and articles, and therefore the true value is probably lower. Interesting thing also is that when I did not prune the DTMs, the number of words was above 80 000. Other than that, the number of collected tweets is as expected. English tweets had the most by large margin. Talking about companies, it is not an surprise that Apple had the most tweets, however it surprised me that Cocacola had less than Boeing. Possible explanation is an collection error, which could be one explanation of the uneven distribution of tweets, where we can observe "plateaus" in figure 1. Another explanation might be marketing campaigns. I believe that the truth lies somewhere between, however bigger influence has the unsatisfactory dataset quality.

| | microsoft | boeing | apple | cocacola | eng tweets |
|---|---|---|---|---|---|
| documents - train | 67 929 | 22 226 | 202 303 | 27 259 | 22 920 000 |
| documents - test | 17 067 | 8 600 | 54 605 | 4 440 | 4 632 500 |
| terms | 1 642 | 1 639 | 1418 | 1 560 | 1195 |

Table 3: Dimensions of document - term matrices in training dataset

### 3.2 Small datasets

After creating DTMs, I decided to run 10-fold crossvalidation, for up to 100 values of penalisation term $\lambda$. The main goal of the crossvalidation was to find such value of $\lambda$ with the smallest Mean Squared Error (MSE). However, to prevent overfitting, following best practices I decided to use such value of $\lambda$ with at least one standard error of MSE value more, than is the minimum. Results can be seen in figure 3. Overall, we can say that we chose models with $\frac{2}{3}$ of the original number words. In the case of Apple, the minimal value was close to the minimal value of lambda, which signalises that by introducing bias to the model, we are not able to reduce a lot of the variance. In the rest of the models we can observe that introduction of the penalisation term worked and we were able to reduce the variance, measured in MSE, in exchange for slightly higher bias. On the note of the MSE interpretation, if we consider that MSE is in the same units as the dependent variable, but squared. For easier interpretation there is sometimes used Root Mean

Squared Error (RMSE), because RMSE is having the same units as the original dependent variable. The results can be seen in table 4. Overall, we can say that these values are quite high, considering that the explained variable attains values between -1, 0 and 1.

| | microsoft | boeing | apple | cocacola |
|---|---|---|---|---|
| RMSE | 0.712 | 0.594 | 0.669 | 0.601 |
| MSE | 0.507 | 0.352 | 0.447 | 0.361 |

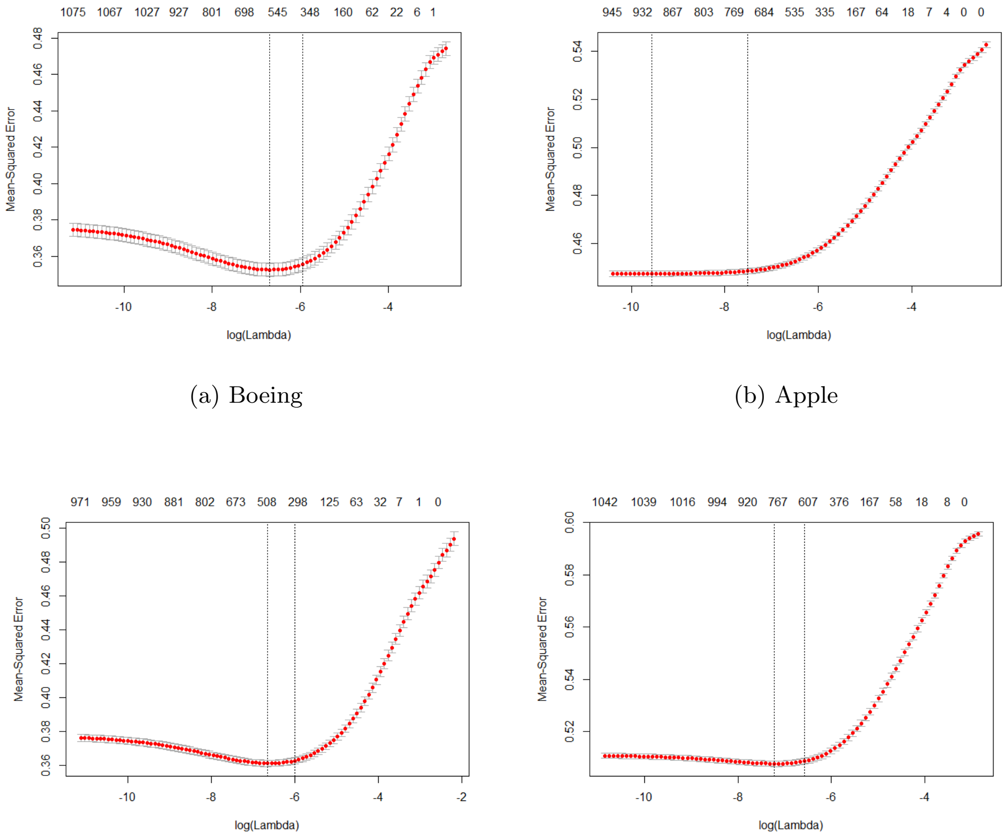
Table 4: Minimal RMSE and MSE of individual multinomial crossvalidated models

### 3.2.1 Generalisation performance of the multinomial model

Overall, we can say that using raw text available on twitter, even though preprocessed via basic document term matrix is not a viable way to predict market movement. In each case, p-value of the hypothesis that relevant model performs better than classifying with the most frequent one is greater than 0.5 (and sometimes is even 1), therefore we cannot reject the null hypothesis. See the table 5 or Appendix B for results. However, confusion matrix provides us with other interesting numbers. In the case of Boeing, we can observe that the classifier was not absolutely random when classifying tweets with negative effect, what can be seen that balanced accuracy was 0.579 for class "-1". Apart from that, all these values are not much different from random value assessing or classifying all values with the most frequent one. On the other hand, excluding the results from Coca Cola, we can observe that the classifier does not even try to attach value for the positive class, however it at least tries to assess negative values. Therefore, I decided to run binomial models and tried to predict which tweets will be negative.

### 3.2.2 Binomial models and their generalisation performance

Confusion tables from multinomial models led me to the conclusion, that classifier guesses for the negative values, however it did not even try to predict the positive ones. Therefore, I decided to run binomial logistic regres-



(a) Boeing

(b) Apple

(c) CocaCola

(d) Microsoft

Figure 3: MSE minimalisation using crossvalidation in multinomial logit lasso model. Lower = better. We can observe two dotted lines in each graph. First line from the right shows the minimal value of the MSE, second line is such value of $\lambda$, which value of MSE is at least one standard value above the minimal MSE. Such value was chosen for prediction.

| | microsoft | boeing | apple | cocacola |
|---|---|---|---|---|
| Accuracy | 0.3018 | 0.7088 | 0.5518 | 0.4014 |
| No Information Rate | 0.3755 | 0.7537 | 0.569 | 0.4998 |
| Balanced accuracy* | 0.4916 | 0.579 | 0.500 | 0.497 |

Table 5: Multinomial models and information from confusion table.I tried to predict whether we can identify tweets containing market relevant information, which would cause significantly positive or negative yields. We can see that in every case the estimated Accuracy is lower than the No Information Rate. In the case of balanced accuracy I decided to report the number for the class "-1", which indicated when the market yields were significantly negative.

sion hoping to find interesting relationship between datasets and significantly negative yields.

You can see the results in table 6 or Appendix C. In the crossvalidation, I decided to use AUC maximalisation. and we can observe that the models became overfitted, in several cases leading to AUC of 0.9. But in the prediction, the results were not different from the multinomial case. However, interesting is again the case of Boeing, where we can observe balanced accuracy of 0.58, an important measure as it takes into account that the classes were not balanced. This led me to investigate the case of Boeing more deeply.

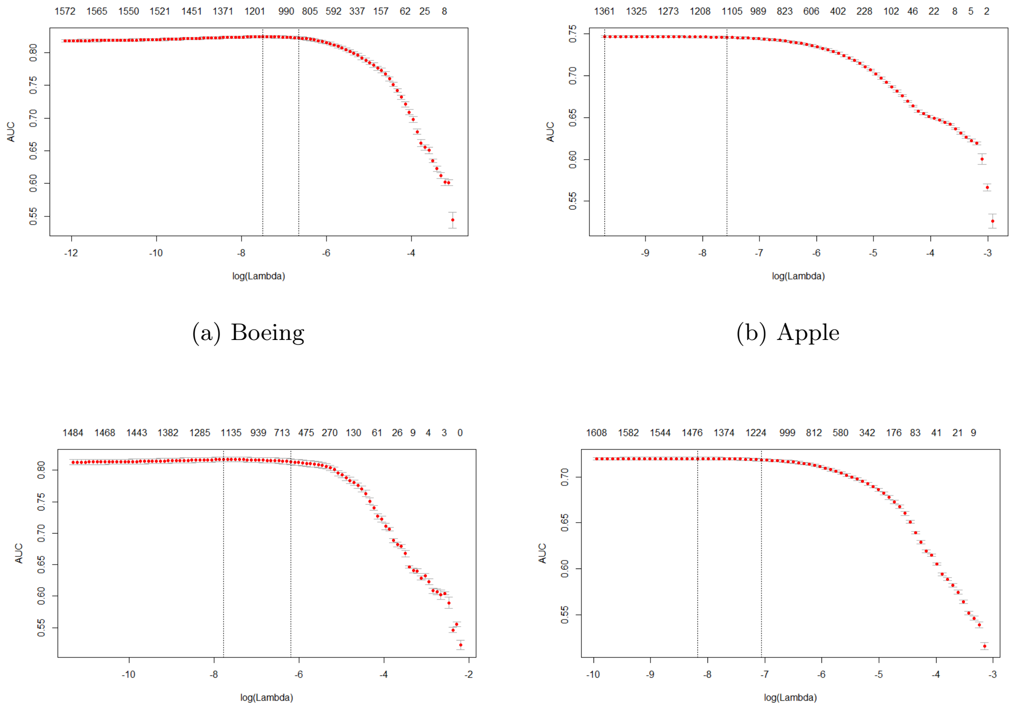
| | microsoft | boeing | apple | cocacola |
|---|---|---|---|---|
| Accuracy | 0.532 | 0.888 | 0.605 | 0.806 |
| No Information Rate | 0.729 | 0.946 | 0.616 | 0.898 |
| Balanced accuracy | 0.494 | 0.580 | 0.502 | 0.499 |

Table 6: Binomial models and information from confusion table. I tried to predict whether we can identify tweets containing market relevant information, which would cause significant loss. We can see that in every case the estimated Accuracy is lower than the No Information Rate.

### 3.2.3 Investigating the case of Boeing and finding market relevant words

Firstly, in the observed period, I have found several stories with potential negative or ambivalent impact at the Boeing share price:

- On 28. 3. 2017, Boeing airplane in Peru caught fire.



(a) Boeing

(b) Apple

(c) CocaCola

(d) Microsoft

Figure 4: AUC maximalisation using crossvalidation in binomial logit lasso model. Higher = better. We can observe two dotted lines in each graph. First line from the right shows the maximal value of the AUC, second line is such value of $\lambda$, which value of AUC is at least one standard value below the maximal AUC. Such value was chosen for prediction.

- On 17. 3. 2017, Boeing presents plans for layoffs
- On 5. 4. 2017, Boeing sealed the $3 bill. deal with Iranian air company.

Secondly, I decided to look deeply in the binomial model and observe the top 15 words with highest effect, out of 900 words with any effect. These results can be seen in table 7. Thirdly, I decided to use Google and Google news and look through all these words, together with the word "boeing". I also looked through Twitter as well. I tried to identify the main context in which the word was used and wrote down in table 8.

| | word | coefficient |
|---|---|---|
| 1 | warns | 4.63 |
| 2 | blazing | 3.69 |
| 3 | lushaviation | 3.58 |
| 4 | hovgeeokr3 | 3.30 |
| 5 | jdam | 3.16 |
| 6 | 90 | 2.97 |
| 7 | tayyabaumar | 2.66 |
| 8 | 74wdgouvuh | 2.47 |
| 9 | oikhx0sehu | 2.36 |
| 10 | newboeingtx | 2.30 |
| 11 | uso | 2.18 |
| 12 | ttb | 2.05 |
| 13 | ebay | 2.01 |
| 14 | costs | 1.99 |
| 15 | dollars | 1.99 |

Table 7: Words with highest coefficients

Finally, we can identify several problems. First, even though the original post had important information like Korea contract, algorithm identified nonsensical word which cannot be used in the future. Sometimes the algorithm only identified an twitter account. In several cases, algorithm also identified Boeing marketing campaign such as jdam or newboeingtx. This might be market relevant, however I would rather consider it as a noise. Other words also identify only one time events, such as "uso" or "90". Also

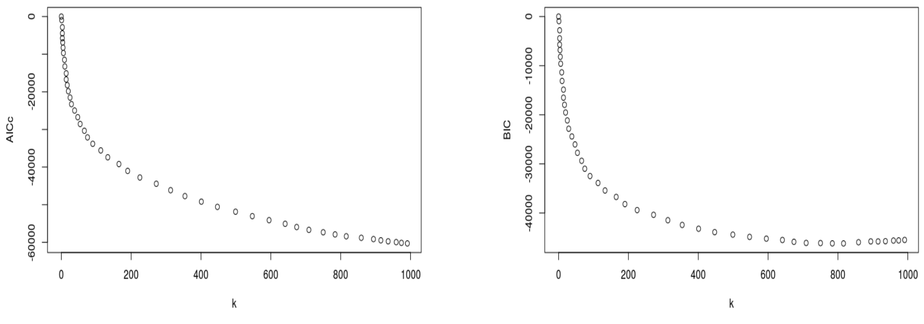
|  | word | explanation |
|---|---|---|
| 1 | warns | Often used with the layoffs |
| 2 | blazing | Often used when planes catch fire |
| 3 | lushaviation | Twitter airplane news account |
| 4 | hovgeeokr3 | Nothing found |
| 5 | jdam | rockets - JDAM Guidance System |
| 6 | 90 | Boeing and 90 comp. urging tax system overhaul |
| 7 | tayyabaumar | Twitter account of aviation student |
| 8 | 74wdgouvuh | https://t.co/74WDGoUvuh - picture |
| 9 | oikhx0sehu | https://t.co/oIKHX0sEhU - Boeing wins korea contract |
| 10 | newboeingtx | New defense airplane, hashtag |
| 11 | uso | VP present at non-profit org. event |
| 12 | ttb | Nothing relevant found |
| 13 | ebay | Might be newsweek article about immigrants |
| 14 | costs | Trump and air force one costs |
| 15 | dollars | Billions of dollars |

Table 8: Identified words with context explanation

Trump's tweet about the costs of Air Force One Boeing, even though very relevant for the market at the time when it was posted, was an old story at the time of dataset collection and fully absorbed by the markets. On the other hand, we can see which words probably might be relevant for markets. If any company warns of any danger, this immediately could be sign for markets to sell. And if there would appear collocation of words such as blazing plane (it has appeared in the news several times throughout past years), that might also be an incentive to sell the stock. However, I have not found any articles in the period when testing sample was collected, where there would be collocation like "boeing warns" or "blazing boeing". Top 20 words for other companies with the highest effect on the negative yields can be seen in Appendix E. We can observe that Boeing was the only one from small companies whose top word was meaningful, and this might be an answer why this model was the most successful of them all.

## 3.3 "All English tweets" dataset and problems of Big Data analysis

Overall, work with the big dataset (consisting of sample from English tweets) was tedious, as everything took a lot of time. While applying the exactly same approach as for the smaller datasets, I used virtual machine with 16 cores and 112 GB of RAM memory. I still was not able to calculate the crossvalidation, as it takes too much memory. Therefore, I decided to ease the parameters. I did not use 10 fold crossvalidation, but only 3 fold. I decided also to lower the number of calculated lambdas to 10 only, along with other parameters for faster computing. I also used binomial model only. But the result was the same and after several hours I ran out of memory again. In the end I ended up calculating model for 100 different values of lambda on data collected in the period between 8th and 29th of March. Afterwards, I used AICc and BIC to choose the model with the lowest expected prediction error. AICc and BIC values can be seen in figure 5. AICc prefered full model with all variables, BIC prefered model with less variables (around 800 words), however it seems that in both cases it seems that introducing bias does not reduce variance by much. Anyway, I decided to try both chosen models.



(a) AICc

(b) BIC

Figure 5: Akaike and Bayesian Information Criteria for binomial model constructed on "All English Tweets" dataset. Lower = better.

Afterwards, I used the models to predict the values from the test dataset, with tweets collected in the period between 29th of March, 2017 and 2nd April, 2017. Results can be seen in table 9 or Appendix D. We can see that there is no difference between models chosen by AICc and BIC and that models predicted very few values. Balanced accuracy is exactly 0.5 in both cases. I believe that these results indicate only that the share of market relevant tweets is close to zero, and that in order to extract market relevant information, using the whole dataset does not make any sense.

|  | AICc | BIC |
|---|---|---|
| Accuracy | 0.4941 | 0.4941 |
| No information rate | 0.5059 | 0.5059 |
| Balanced Accuracy | 0.5 | 0.5 |

Table 9:

Information from confusion table made on "All English tweets" dataset. We can observe that choosing different value of $\lambda$ made no difference. Again, Accuracy of the model is lower, than the No Information Rate

I decided to do ex-post analysis of the market relevant words in line with Bollen, Mao and Zeng (2011) or Mittal and Goel (2012). It seems intuitive, that as these studies worked only with words connected with feelings, I expect to see any words associated with feelings in the top 20 words with highest coefficents. However, we can observe in table 10, that the main negative driver of the DJIA index was probably American health care reform, and we cannot see many words, which would have any relevance to the feelings. It is questionable whether "weather" or "dating" can be connected with negative feelings, especially negative ones. Moat, Olivola et al. (2016) on the other hand identified word "color" as significant for prediction of the markets using Google trends, and we can see word "red" on the 6th position, but I would rather expect this as an influence of introduction of the red iPhone by Apple. But we still can also observe several words which could be used in the sentence with word "debt", e.g. "million" and also the discussion around the health care reform revolved around the government debt as well.

|  | Apple | Boeing | Coca Cola | Microsoft | EN tweets |
|---|---|---|---|---|---|
| 1 | hxn8fyl1rb | warns | 0r6zelcwfb | pzet8hwv5f | Obamacare |
| 2 | pdf | blazing | sspnwdfoke | taps | Ryan |
| 3 | veagan | lushaviation | blessedwithaheart | ikhqreni8s | bill |
| 4 | overturns | hovgeeokr3 | 170323 | yaboybillnye | Republicans |
| 5 | uxawegpxae | jdam | thankyoucoke | gossipgriii | 14 |
| 6 | shawty | 90 | publicly | rodriguezthagod | red |
| 7 | alexandracarmil | tayyabaumar | power | evolution | April |
| 8 | snolehdi5v | 74wdgouvuh | keemstar | historic | 22 |
| 9 | 887 | oikhx0sehu | labeling | newsnight | 24 |
| 10 | incredible | newboeingtx | l0lzyqlmlk | ndv8hcdcnn | million |
| 11 | 97zszf72ak | uso | shutdown | corpus | healthcare |
| 12 | ios11 | ttb | employed | components | FBI |
| 13 | bigdata | ebay | a26xp2pmgg | y0kya4rx0z | South |
| 14 | nook | costs | 46uozyz5u7 | fan | weather |
| 15 | turkish | dollars | refresh | division | played |
| 16 | government | 747sp | eyes | jllpsein28 | EU |
| 17 | simply | passing | sargon_of_akkad | rollbacks | YouKnowYouLoveThem |
| 18 | fr | deep | drvandanashiva | capcom | dating |
| 19 | local | qqltyyj03t | ftifhsigmp | 53xwufcqka | exactly |
| 20 | debut | jonostrower | h7nmvcluhl | o40fpsaaye | water |

Table 10: Top 20 most significant words from binomial logistic model explaining significantly negative yields.

## Conclusion

In this thesis I examined the relationship between social networks, especially Twitter and tried to invent new method where I tried to link tweets, transformed to Document - Term Matrix to the market movement. Overall, used method did not provide useful means to develop efficient market strategy. In the following section I discuss the main reasons.

The biggest drawback of this thesis is dubious dataset quality. Main cause is probably the Terms of Service of Twitter, which restricts to share large and complete datasets. And thus to bypass this, I had to use my own resources. I was able to configure virtual server and also to write such script, which downloaded desired tweets, and I set it to run every 5 minutes. However, I am still unsure about the reasons, why it had stopped several times. My personal guess is that the Twitter "signed me off", that means that it required to log in to the Twitter with the provided API keys once again, however the download script was not written to handle this action. From the figure 1 we can observe that the most doubtful dataset is probably the Coca Cola one, where the number of downloaded tweets is unexpectedly lower even than in the case of Boeing. However I believe that the rest of the datasets allowed me to provide insightful analysis, and especially the "All English tweets" dataset. We can also observe in this case that the identified words with highest coefficients make the most sense. Still, for future analysis, I will definitely write such search script which will also leave log, how the data were downloaded and whether there were any errors. If yes, it would send me message so I could quickly correct that. Also regarding the dataset, I also believe that for the quality of the analysis it would be good to have longer period of tweets collection, maybe up to year. While speaking of the dataset, it is also important to state that free Twitter API do not guarantee that when user performs a search query he gets 100% of the messages. This is available only through Twitter Firehose API, which is expensive.

I chose to work with text directly as I believe that this field has huge potential, however the only thing we need is to develop better algorithms so the computers could understand the text better. But in this thesis we can observe that to understand overall market movements, the raw word frequency using Bag-of-words model is simply not enough. And to be honest, we can observe that it also is not enough to capture the meaning of the sentence at all. However, the Google research team around Tomas Mikolov had recently published *word2vec* algorithm, which creates vector space of words. I have already used the algorithm in the Data Science project before and in my opinion its results for capturing relationships between the words were much better. However I decided not to use it in this thesis, as I had focused at the individual words. Deciding which word has the biggest influence would not be possible using word2vec algorithm. However, if we omit the requirement to have interpretable model, using word2vec model would be better choice.

Speaking of deciding which word has the biggest influence, in this thesis I had worked for the first time with penalised logistic regression. It seemed as a good fit as we only needed to know which word has the biggest influence overall, not the effect on the market, which would be questionable anyway. Therefore I think that introducing bias helped a lot, however as said before, the relationship between Twitter word frequency and market is definitely very weak. But I believe that there was another approach possible in this thesis, and that we could aggregate DTM by the half - hour and try to determine which words predict the market (kind of the opposite relationship - instead of making financial dataset bigger to fit DTM, I could make DTM smaller to fit the financial dataset). However, the main drawback of this approach is that then we would have around 200 degrees of freedom, however 2000 of the explaining variables. Regularisation and crossvalidation are useful methods to bypass this problem. Using lasso regression, we would be able identify at most the *df* of the selected variables, and using ridge regression, we would identify that all of them are useful in the model. Therefore, to obtain the most meaningful results, I would use *elastic-net* penalty, already presented in equation 5. To determine which parameter $\alpha$ would be the best for the most meaningful results remains as an incentive for further work.

I would like to mention pruning as well. In this thesis I omitted around 30 most common words or character strings common in web address. I decided to leave the strings of the individual web addresses as I thought that they actually also contain interesting information, but retrospectively, I think that this was a mistake and that whole address should have been omitted. I also pruned the tweets beforehand by using the Twitter search algorithm. If we consider the case of Boeing, I believe that we cannot assume that all tweets that which could influence the Boeing share price contain word "Boeing" or hashtag "#boeing". And vice versa, if we search for the tweets which really contain these words, not all of the tweets could really affect the share price. In fact, most of them just contain pictures of planes. Therefore, in the next work I would pay far bigger attention to pruning, both of the words as well as the individual tweets. The question remains, how? Interesting idea is that we could be investigating individual company stock, and continuously download its tick data. Whenever the stock price significantly moves, we would download the tweets which were posted about the industry in $n$ minutes before and $m$ minutes after. This also remains as an incentive for further work.

On a positive note, what seems interesting for me is that this approach was able to identify words connected with significant market events, however most of these words were "one trick ponies" - usually they were mentioned only once, such as in the title of the article, or with one individual event. This is the main contribution of this thesis, as it seems to me that this could help with identification of market relevant information, however only *ex-post*. Question is whether the algorithm would do better job than person, who is trading with such stocks and pays close attention to every available information about the company, on the other hand, if it would be automated together with e.g. Google search, this might be good way to obtain thorough but still quick summary.

To answer the original research question, I believe that this method does have potential to obtain such list of words, which could indicate market movement, however it needs further research with more careful choice of the examined tweets, and using other, more sophisticated methods for text processing than DTM. I believe that using this exact method, we are able to identify the words, which were used in the stories which led to market movement, however this analysis is useful only ex-post and cannot be used for forecasting.

# References

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

Bollen, Johan, Huina Mao and Xiaojun Zeng (2011). 'Twitter mood predicts the stock market'. In: *Journal of computational science* 2.1, pp. 1–8.

Eichstaedt, Johannes C et al. (2015). 'Psychological language on Twitter predicts county-level heart disease mortality'. In: *Psychological science* 26.2, pp. 159–169.

Fama, Eugene F (1965). 'The behavior of stock-market prices'. In: *The journal of Business* 38.1, pp. 34–105.

— (1970). 'Efficient capital markets: A review of theory and empirical work'. In: *The journal of Finance* 25.2, pp. 383–417.

Fortmann-Roe, Scott (2012). 'Understanding the bias-variance tradeoff'. In:

Friedman, Jerome, Trevor Hastie and Robert Tibshirani (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. New York, NY: Springer-Verlag New York.

— (2010). 'Regularization Paths for Generalized Linear Models via Coordinate Descent'. In: *Journal of Statistical Software* 33.1, pp. 1–22. URL: http://www.jstatsoft.org/v33/i01/.

Gayo-Avello, Daniel (2013). 'A meta-analysis of state-of-the-art electoral prediction from Twitter data'. In: *Social Science Computer Review* 31.6, pp. 649–679.

Gerber, Matthew S (2014). 'Predicting crime using Twitter and kernel density estimation'. In: *Decision Support Systems* 61, pp. 115–125.

Jurafsky, Dan and James H Martin (2014). *Speech and language processing*. Vol. 3. Pearson.

Kohavi, R and F Provost (1998). 'Confusion matrix'. In: *Machine learning* 30.2-3, pp. 271–274.

Kristoufek, Ladislav, Helen Susannah Moat and Tobias Preis (2016). 'Estimating suicide occurrence statistics using Google Trends'. In: *EPJ Data Science* 5.1, p. 32.

Lazer, David et al. (2014). 'The parable of Google Flu: traps in big data analysis'. In: *Science* 343.6176, pp. 1203–1205.

Malkiel, Burton G (1989). 'Efficient market hypothesis'. In: *The New Palgrave: Finance. Norton, New York*, pp. 127–134.

— (2003). 'The efficient market hypothesis and its critics'. In: *The Journal of Economic Perspectives* 17.1, pp. 59–82.

Mittal, Anshul and Arpit Goel (2012). 'Stock prediction using twitter sentiment analysis'. In: *Standford University, CS229 (2011 http://cs229. stanford. edu/proj2011/GoelMittal-StockMarketPredictionUsingTwitterSentimentAnalysis. pdf)* 15.

Moat, Helen Susannah, Chester Curme et al. (2013). 'Quantifying Wikipedia usage patterns before stock market moves'. In: *Scientific reports* 3.

Moat, Helen Susannah, Christopher Y Olivola et al. (2016). 'Searching Choices: Quantifying Decision-Making Processes Using Search Engine Data'. In: *Topics in cognitive science* 8.3, pp. 685–696.

Nation, Paul and Robert Waring (1997). 'Vocabulary size, text coverage and word lists'. In: *Vocabulary: Description, acquisition and pedagogy* 14, pp. 6–19.

Preis, Tobias and Helen Susannah Moat (2014). 'Adaptive nowcasting of influenza outbreaks using Google searches'. In: *Royal Society open science* 1.2, p. 140095.

Preis, Tobias, Helen Susannah Moat and H Eugene Stanley (2013). 'Quantifying trading behavior in financial markets using Google Trends'. In:

Timmermann, Allan and Clive WJ Granger (2004). 'Efficient market hypothesis and forecasting'. In: *International Journal of forecasting* 20.1, pp. 15–27.

Tumasjan, Andranik et al. (2011). 'Election forecasts with Twitter: How 140 characters reflect the political landscape'. In: *Social science computer review* 29.4, pp. 402–418.

# List of tables and figures

## List of Figures

1 Number of obtained words for individual companies. . . . . . 13

2 Number of obtained words for sample from all tweets . . . . 13

3 Mean squared error crossvalidation . . . . . . . . . . . . . . . 22

4 AUC logistic model crossvalidation . . . . . . . . . . . . . . . 24

5 AIC and BIC of binomial models constructed on "All English Tweets" dataset . . . . . . . . . . . . . . . . . . . . . . . . 27

## List of Tables

1 Illustration of document term matrix for single Donald Trump's tweet. . . . . . . . . . . . . . . . . . . . . . . . . . . . 10

2 Illustration of pruned document term matrix. . . . . . . . . 11

3 Dimensions of document - term matrices in training dataset 20

4 Minimal RMSE and MSE of individual multinomial crossvalidated models . . . . . . . . . . . . . . . . . . . . . . . . . . 21

5 Multinomial models and information from confusion table. . 23

6 Binomial models and information from confusion table. . . 23

7 Words with highest coefficients . . . . . . . . . . . . . . . . 25

8 Identified words with context explanation . . . . . . . . . . . 26

9 Information from confusion table made on "All English tweets" dataset. . . . . . . . . . . . . . . . . . . . . . . . . . . . 28

10 Top 20 most significant words from binomial logistic model explaining significantly negative yields. . . . . . . . . . . . . 29

# Appendix A

It is a very natural question to ask for standard errors of regression coefficients or other estimated quantities. In principle such standard errors can easily be calculated, e.g. using the bootstrap.

Still, this package deliberately does not provide them. The reason for this is that standard errors are not very meaningful for strongly biased estimates such as arise from penalized estimation methods. Penalized estimation is a procedure that reduces the variance of estimators by introducing substantial bias. The bias of each estimator is therefore a major component of its mean squared error, whereas its variance may contribute only a small part.

Unfortunately, in most applications of penalized regression it is impossible to obtain a sufficiently precise estimate of the bias. Any bootstrap-based calculations can only give an assessment of the variance of the estimates. Reliable estimates of the bias are only available if reliable unbiased estimates are available, which is typically not the case in situations in which penalized estimates are used.

Reporting a standard error of a penalized estimate therefore tells only part of the story. It can give a mistaken impression of great precision, completely ignoring the inaccuracy caused by the bias. It is certainly a mistake to make confidence statements that are only based on an assessment of the variance of the estimates, such as bootstrap-based confidence intervals do. Reliable confidence intervals around the penalized estimates can be obtained in the case of low dimensional models using the standard generalized linear model theory as implemented in lm, glm and coxph. Methods for constructing reliable confidence intervals in the high-dimensional situation are, to my knowledge, not available.

*Jelle Goeman, Ph.D., author of the package "penalized"*

## Appendix B - Confusion matrices of Multinomial logit

### Microsoft Confusion Matrix

```
Confusion Matrix and Statistics

          Reference
Prediction   -1    0    1
        -1 2843 3711 4162
        0  1751 2285 2223
        1    24   45   23

Overall Statistics
               Accuracy : 0.3018
                 95% CI : (0.2949, 0.3088)
    No Information Rate : 0.3755
    P-Value [Acc > NIR] : 1

                  Kappa : 1e-04
 Mcnemar's Test P-Value : <2e-16

Statistics by Class:

                     Class: -1 Class: 0 Class: 1
Sensitivity             0.6156   0.3782 0.003589
Specificity             0.3676   0.6396 0.993527
Pos Pred Value          0.2653   0.3651 0.250000
Neg Pred Value          0.7205   0.6525 0.623859
Prevalence              0.2706   0.3540 0.375461
Detection Rate          0.1666   0.1339 0.001348
Detection Prevalence    0.6279   0.3667 0.005391
Balanced Accuracy       0.4916   0.5089 0.498558
```

**Boeing Confusion Matrix**

```
Confusion Matrix and Statistics

          Reference
Prediction   -1    0    1
        -1  107  475  106
        0   358 5977 1535
        1     0   30   12

Overall Statistics

               Accuracy : 0.7088
                 95% CI : (0.6991, 0.7184)
    No Information Rate : 0.7537
    P-Value [Acc > NIR] : 1

                  Kappa : 0.0453
 Mcnemar's Test P-Value : <2e-16

Statistics by Class:

                     Class: -1 Class: 0 Class: 1
Sensitivity            0.23011   0.9221 0.007260
Specificity            0.92858   0.1062 0.995682
Pos Pred Value         0.15552   0.7595 0.285714
Neg Pred Value         0.95475   0.3082 0.808250
Prevalence             0.05407   0.7537 0.192209
Detection Rate         0.01244   0.6950 0.001395
Detection Prevalence   0.08000   0.9151 0.004884
Balanced Accuracy      0.57934   0.5142 0.501471
```

### CocaCola Confusion Matrix

```
Confusion Matrix and Statistics

          Reference
Prediction   -1    0    1
        -1    5   30   34
        0   446 1732 2140
        1     2    6   45

Overall Statistics

               Accuracy : 0.4014
                 95% CI : (0.3869, 0.4159)
    No Information Rate : 0.4998
    P-Value [Acc > NIR] : 1

                  Kappa : 0.0108
 Mcnemar's Test P-Value : <2e-16

Statistics by Class:

                     Class: -1 Class: 0 Class: 1
Sensitivity           0.011038  0.97964  0.02028
Specificity           0.983948  0.03219  0.99640
Pos Pred Value        0.072464  0.40111  0.84906
Neg Pred Value        0.897506  0.70492  0.50444
Prevalence            0.102027  0.39820  0.49977
Detection Rate        0.001126  0.39009  0.01014
Detection Prevalence  0.015541  0.97252  0.01194
Balanced Accuracy     0.497493  0.50591  0.50834
```

**Apple Confusion Matrix**

```
Confusion Matrix and Statistics

          Reference
Prediction    -1     0     1
        -1  1823  2680   235
        0  19012 28305  2320
        1    142    84     4

Overall Statistics

               Accuracy : 0.5518
                 95% CI : (0.5476, 0.556)
    No Information Rate : 0.569
    P-Value [Acc > NIR] : 1

                  Kappa : 0.0024
 Mcnemar's Test P-Value : <2e-16

Statistics by Class:

                     Class: -1 Class: 0  Class: 1
Sensitivity            0.08690  0.91104 1.563e-03
Specificity            0.91332  0.09364 9.957e-01
Pos Pred Value         0.38476  0.57024 1.739e-02
Neg Pred Value         0.61590  0.44364 9.530e-01
Prevalence             0.38416  0.56898 4.686e-02
Detection Rate         0.03339  0.51836 7.325e-05
Detection Prevalence   0.08677  0.90902 4.212e-03
Balanced Accuracy      0.50011  0.50234 4.986e-01
```

## Appendix C - Confusion matrices of binomial logit

### Microsoft

```
Confusion Matrix and Statistics

          Reference
Prediction    0    1
         0 7196 2728
         1 5253 1890

               Accuracy : 0.5324
                 95% CI : (0.5249, 0.5399)
    No Information Rate : 0.7294
    P-Value [Acc > NIR] : 1

                  Kappa : −0.0108
 Mcnemar's Test P-Value : <2e−16

            Sensitivity : 0.5780
            Specificity : 0.4093
         Pos Pred Value : 0.7251
         Neg Pred Value : 0.2646
             Prevalence : 0.7294
         Detection Rate : 0.4216
   Detection Prevalence : 0.5815
      Balanced Accuracy : 0.4937

       'Positive' Class : 0
```

### Boeing

```
Confusion Matrix and Statistics

          Reference
Prediction    0    1
         0 7530  356
         1  605  109

               Accuracy : 0.8883
                 95% CI : (0.8814, 0.8948)
    No Information Rate : 0.9459
    P-Value [Acc > NIR] : 1

                  Kappa : 0.1278
 Mcnemar's Test P-Value : 1.244e-15

            Sensitivity : 0.9256
            Specificity : 0.2344
         Pos Pred Value : 0.9549
         Neg Pred Value : 0.1527
             Prevalence : 0.9459
         Detection Rate : 0.8756
   Detection Prevalence : 0.9170
      Balanced Accuracy : 0.5800

       'Positive' Class : 0
```

**Apple**

```
Confusion Matrix and Statistics

          Reference
Prediction     0     1
         0 31762 19713
         1  1866  1264

               Accuracy : 0.6048
                 95% CI : (0.6007, 0.6089)
    No Information Rate : 0.6158
    P–Value [Acc > NIR] : 1

                  Kappa : 0.0057
 Mcnemar's Test P–Value : <2e−16

            Sensitivity : 0.94451
            Specificity : 0.06026
         Pos Pred Value : 0.61704
         Neg Pred Value : 0.40383
             Prevalence : 0.61584
         Detection Rate : 0.58167
   Detection Prevalence : 0.94268
      Balanced Accuracy : 0.50238

       'Positive' Class : 0
```

# Appendix D

## Sample from all tweets

### AICc minimal value

```
Confusion Matrix and Statistics

          Reference
Prediction       0       1
         0 2289065 2343373
         1      35      27

               Accuracy : 0.4941
                 95% CI : (0.4937, 0.4946)
    No Information Rate : 0.5059
    P-Value [Acc > NIR] : 1

                  Kappa : 0
 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 1.000e+00
            Specificity : 1.152e-05
         Pos Pred Value : 4.941e-01
         Neg Pred Value : 4.355e-01
             Prevalence : 4.941e-01
         Detection Rate : 4.941e-01
   Detection Prevalence : 1.000e+00
      Balanced Accuracy : 5.000e-01

       'Positive' Class : 0
```

**BIC minimal value**

```
Confusion Matrix and Statistics

          Reference
Prediction       0       1
         0 2289068 2343377
         1      32      23

               Accuracy : 0.4941
                 95% CI : (0.4937, 0.4946)
    No Information Rate : 0.5059
    P-Value [Acc > NIR] : 1

                  Kappa : 0
 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 1.000e+00
            Specificity : 9.815e-06
         Pos Pred Value : 4.941e-01
         Neg Pred Value : 4.182e-01
             Prevalence : 4.941e-01
         Detection Rate : 4.941e-01
   Detection Prevalence : 1.000e+00
      Balanced Accuracy : 5.000e-01

       'Positive' Class : 0
```