Norwegian School of Economics
Bergen, Spring 2018

The predictive power of Earnings Conference Calls

Predicting Stock Price Movement With Earnings Call Transcripts

Lars Erik Solberg

Jørgen Karlsen

Supervisor: Walter Pohl

Master's thesis in Financial Economics

Norwegian School of Economics

Abstract

Earnings conference calls are considered a valuable text based information source for investors. This paper investigates the possibility to predict the direction of stock prices by analyzing the transcripts of earnings conference calls. The paper investigates 29 339 different earnings call transcript from 2014 to 2017 and classify the individual documents to either be part of class *up* or *down*. Four different machine learning algorithms are used to classify and predict based on the bag of words method. These machine learning algorithms are Naive Bayes, Logistic regression with lasso regularization, Stochastic Gradient Boosting, and Support Vector Machine. All models are compared to a benchmarks based on S&P 500. The model with best performance is logistic regression with a classification error of 43,8%. In total, 2 of 4 models beats the benchmark significantly, namely logistic regression and gradient boosting. With these results, the paper concludes that earnings calls contain predicting power for next day's stock price direction.

Preface

This master thesis is a part of the master degree from NHH Norwegian School of Economics. It was written the spring semester of 2018.

This thesis combines the fields of finance, textual analysis and machine learning to see if transcribed earnings conference calls can predict stock price movement. The thesis was motivated by an interest developed through different courses during our years at NHH.

We would like to thank our supervisor Walter Pohl and Maximilian Rohrer which supported us with valuable input.

Contents

List of Figures

List of Tables

1 Introduction

The last half century have seen an exponential increase in computer power.[1] Technological development has led to new and powerful ways to explore data. At the same time, more data is generated and stored to be explored every day. International Data Corporation forecasts that by 2025 the global datasphere will grow to 163 zettabytes (IDC, 2017). In combination with this, new technology makes data more accessible and provides possibilities to develop new insights.

Within the world of finance, textual analysis is an emerging area due to technological development (Guoa, Shib, and Tua, 2016). News articles, social media, Securities and Exchange Commission (SEC) filings and earnings conference call transcripts are all text based financial sources. These information sources might provide interesting findings and insights.

A relatively unexplored source of information in the finance industry is the earnings conference call. Tasker (1998) argues that quarterly conference calls have certain advantages over other company disclosure metrics. She emphasizes that some types of information about a business is not easily conveyed through traditional financial reporting channels e.g. helping employees acquire a new skill. Moreover, she provides evidence that managers of small and medium sized firms provides additional disclosure to shareholders during conference calls.

The earnings conference call is usually held within few hours to a day following the publishing of the earnings announcement press release (Kimbrough, 2005). It is common for an earnings conference call to use the following structure. Firstly, company representatives holds a speech about the foregoing quarter and the prospects for the future.

[1] Moore's and Koomey's law.

Subsequently the representatives answers questions from the participants in a questions and answers session (Q&A). Generally, the representatives will be different managers, while participants will be large institutional investors or analysts.

As a response to Regulation Fair Disclosure, passed by the SEC in 2000, the earnings conference calls became accessible to the public. The regulation provides that when issuing disclosed material to any given person or professional institution, the issuer must also make the information available to the public. This resulted in earnings calls being transcribed. These transcripts are now subject to researchers trying to find new insights, hidden value or predictive power.

Bowen, Davis, and Matsumoto (2001) discovers evidence that conference calls are a valuable indication for future earnings by examining analysts' forecasting accuracy on next quarter's earnings. They document increased prediction accuracy when an earnings announcement are accompanied by a conference call. Furthermore, Kimbrough (2005) reports how earnings announcements along with conference calls leads to decreased post-earnings-announcement-drift which is explained by the more profound information that analysts can derive from the Q&A-session.

The linguistic content of conference calls has for a long time been subject of discussion for researchers. In his review of disclosure literature, Core (2001) conjectures that one should borrow from fields like computer science, linguistics and artificial intelligence when analyzing company disclosure conveyed in natural language like conference calls. He further argued that this would open for easier generalizations about the tone and sentiment of company disclosure.

This is consistent with Matsumoto, Pronk, and Roelofsen (2011). They confirm the value of the incremental information which follows an earnings conference call, and highlight that the Q&A session has more informational content. They find that when

firm performance is poor, the management use more non-financial and future-oriented language in the prepared presentation. In those circumstances, they demonstrate that the associated Q&A session is relatively more informative because more questions are being asked by the participants.

Price, Doran, Peterson, and Bliss (2012) points out that managers possess superior information about future prospects compared to investors, and that this manifests itself in the linguistic tone of conference calls. By using textual data analysis, they examine the incremental informativeness of quarterly earnings calls and find that the linguistic conference call tone is a significant predictor of abnormal returns and trading volumes.

Larcker and Zakolyukina (2012) revealed that by using predictive models it is possible to classify conference calls as being either ”deceptive” or ”trustful”. Based on linguistic features, they estimate a model that is considerably better than guessing whether the CEO or CFO is being deceitful or trustful. They found that deceitful CEO's use more extreme positive emotion words, and deceitful CFO's uses fewer self-references.

This short literature review on earnings conference calls demonstrate that there are valuable incremental information in the earnings calls transcripts. Previous research have in different ways extracted valuable information and suggest that it would be beneficial to use computer science when analyzing results from earnings calls. This master thesis will try to benefit from these previous researches and their findings.

The research question of this thesis is to see if transcripts of companies' earnings conference calls can be used to correctly predict the stock price movement the next day. The prediction will be a classification problem, where four different machine learning algorithms will use earnings call transcripts. The transcripts will be from companies listed on either New York Stock Exchange (NYSE) or National Association of Securities Dealers Automated Quotations (NASDAQ).

This is an interesting and important topic due to various reasons. Firstly, to our knowledge, analyzes of earnings call transcript has not mainly been used to predict stock price movement on NASDAQ and NYSE listings. Secondly, a successfully developed model with high performance would reduce investor costs when analyzing companies. Thirdly, models explored by this thesis are not only classifiers that predict next day's price movement, they also have some applications in regards to classifying positive and negative linguistic content. Since the amount of text within the financial industry is of an large proportion, finding sophisticated ways to exploit this could be of great value to the industry. Lastly, this thesis might be an inspiration to others, by illustrating how different emerging fields such as textual analysis and machine learning can be applied to the more traditional field of finance.

This thesis consists of 6 sections. The rest of the paper is structured as follows. Section 2 consists of theory on important aspects to this thesis and relevant literature review. Section 3 presents the methodological choices that are made for this thesis and short discussions on how these choices help achieve the goal of this study. Section 4 provides the results through a descriptive and empirical analysis. Section 5 is a discussion regarding the results. Section 6 concludes and suggest further research within this topic.

2 Theory and Literature Review

This section aims to present the reader with a solid understanding on previous research and concepts which are important for this thesis. The first subsection will take a closer look at text mining. The second subsection will dive into the field of text analysis and finance. The third is a part on the different machine learning techniques used in this thesis. The section ends with subsections on machine learning concepts and performance measures.

2.1 Text Mining

Text, reports and articles consists mainly of unstructured data. Unstructured data is data that cannot be connected to anything, it has no recognizable structure. To exemplify, a number by itself provides little information. But when the number is connected to the costs of goods sold, then it becomes useful information.

A text based example on unstructured data can be done with an email. An email usually contains time, date, subject etc. Still, the body of the email remains unstructured. The body does not relate to anything that can be analyzed. This is were the term text mining becomes relevant. With text mining, different techniques can be applied to a text and make the data useful.

Gaikwad, Chaugule, and Patil (2014) defines text mining as a process of extracting useful information and knowledge from text. Some of the different techniques and methods that can be applied are categorization, clustering, sentiment analysis or natural language processing. Using categorization as an example, if sections of texts are to be categorized based on whether or not they are of financial content, the sections would be

given label ”financial” or ”normal” using text mining techniques. After this process, the body of the text can be considered structured.

Figure 1 illustrates generally how text mining is used to develop insights. Firstly, the text miner defines the objective. Then it is necessary to take a look at the nature of the data and assess what form this data is portrayed. Extraction of information is then achieved using various text mining techniques. While the exact techniques depends on the data and the goal, a well thought process will deliver new insight to the text due to its now more structured form.

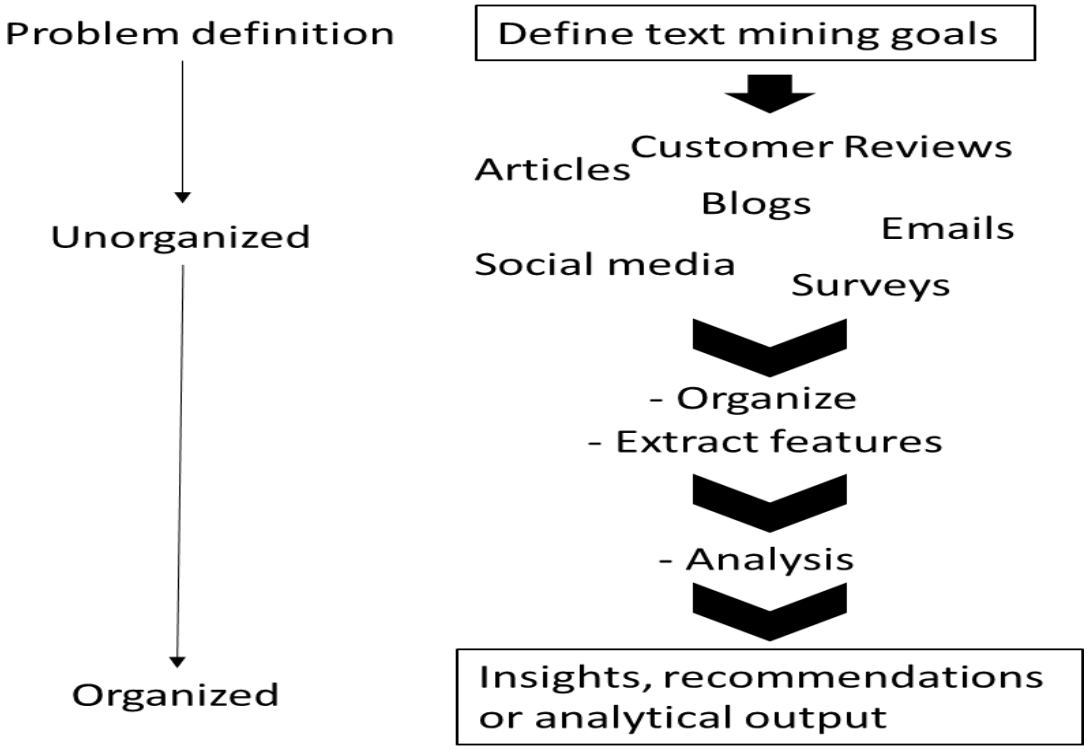


Figure 1: The process of text mining.

A similar, but more specific definition of text mining is given by Kwartler (2017). He defines text mining as ”the ability to take large amounts of unstructured language and

quickly extract useful and novel insights that can affect stakeholder decision-making". With this definition, he is accentuating that text mining is a tool that can help decision makers.

Kwartler (2017) also points out that modern text mining are helping businesses transforming unstructured data, in the form of public opinions from sources such as blogs and customer-reviews, into structured and orderly information sources. Ignoring to mine external text sources, or failing to structurally put text into order can lead to false outcomes of analysis, and will make businesses miss valuable opportunities.

Figure 2 illustrates some of the major fields where text mining are in use today. As an example, this thesis relies heavily on the aspects of text mining dealing with information retrieval and data science to structure transcripts and to derive information for further analysis.

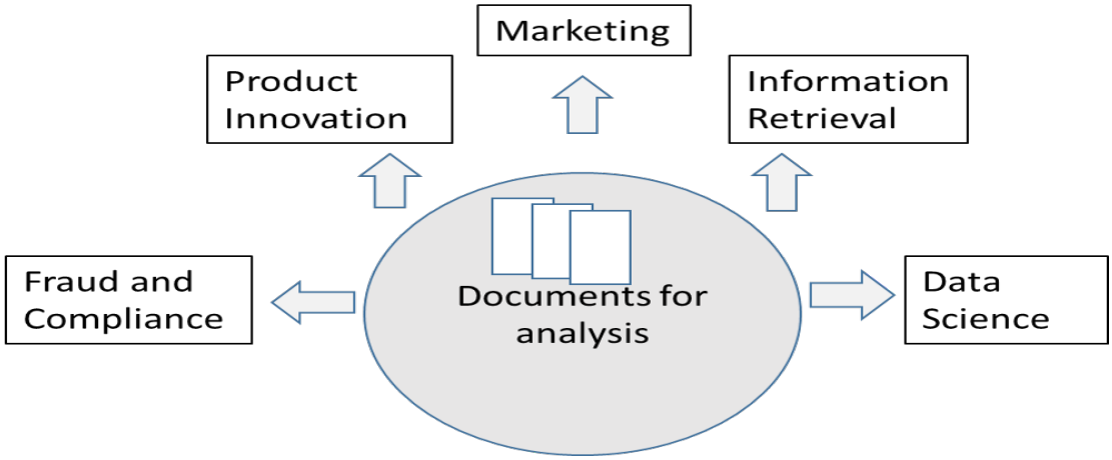


Figure 2: Possible uses of text mining for enterprises.

Techniques of text mining can broadly be divided into two types. These are called *bag of words* and *syntactic parsing*. Both types has their advantages and disadvantages. This

paper will only use and focus on the bag of words approach.

Bag of words treat all words or *group of words* as a unique feature of the document.[2] Word order and grammatical word type are disregarded in a bag of words analysis. Figure 3 presents how a movie review is broken down into a bag of words, where every unique word is a feature.

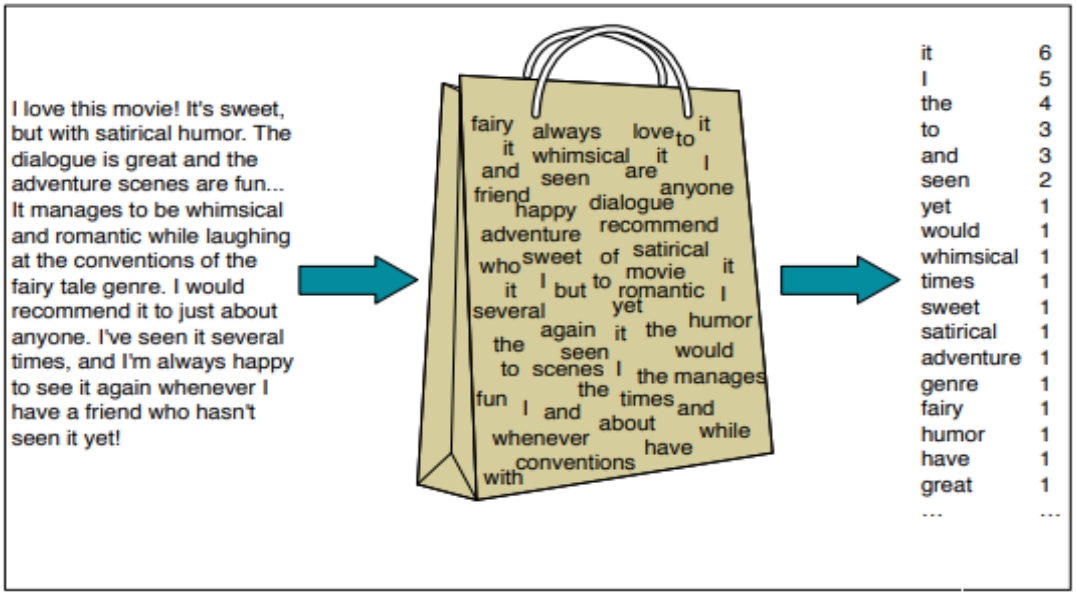


Figure 3: Bag of words representation (Jurafsky and Martin, 2017).

One of the advantages of bag of words is that it is generally not computationally expensive (Loughran and McDonald, 2016). It is also not demanding to organize a corpus for text mining. In other words, analyzes can be done relatively quickly. This is a great advantage when working with text, in which large and time consuming data sets are generated.

Bag of words do also suit machine learning frameworks well. This is because the data are arranged in a matrix of attributes and observations. These matrices are usually

[2]Group of words is often referred to as n-grams.

referred to as document term matrix (DTM) or term document matrix (TDM). The difference between the two matrices dwells on whether one wants the documents to be the columns and words to be the rows and vice versa.

To exemplify the difference between DTM and TDM, consider these two sentences:

- Sentence 1: Financial text contains information.
- Sentence 2: Use R on financial text.

These two sentences represent a corpus and will be organized in the following way as a DTM:

	financial	text	contains	use	R	information	on
Sentence 1	1	1	1	0	0	1	0
Sentence 2	1	1	0	1	1	0	1

Table 1: Example of a document term matrix.

The DTM shows only word counts. The matrix display the word counts as they appeared for the specific sentence. Table 1 exemplifies how the data is organized. From this DTM there could be made a quick assumption based on word frequency. A suggestion is that this corpus is about *financial* and *text*.

Using the same corpus as in the DTM example, a TDM would look like this:

	sentence 1	sentence 2
financial	1	1
text	1	1
contains	1	0
use	0	1
R	0	1
information	1	0
on	0	1

Table 2: Example of a term document matrix.

In this case, there is no real difference between the two matrices. But usually, the choice of DTM or TDM comes down to the objective and analyzing task at hand.

Loughran, McDonald, and Yun (2008) uses bag of word methods to find target phrases. They focus on words as *ethics* and variants of ethics together with words as *corporate responsibility, social responsibility etc.*, in 10-K filings to find out if these terms are associated with sin stocks[3], class action suits and corporate governance measures. They find that firms that asserts more attention to discuss these topics in their 10-K reports are more often referred to as sin stocks. These companies are more likely to be sued and have low corporate governance measures.

2.2 Text Analysis and Finance

Although textual analysis is an up and coming field in accounting and finance, there is a decent amount of research done on the area. Li (2008) provides a survey on ”older” literature regarding textual analysis and discusses this within topics as market efficiency and earnings quality. Another survey by Kearney and Liu (2014) dives into more recent literature in contrast to Li. Their survey emphasizes more on textual sentiment. Das

[3]A sin stock or a sinful stock is publicly traded companies that are considered unethical or immoral.

(2014) surveys the technology and empiricism of text analyzes in finance. His monograph is useful for anyone entering the field and comes with code snippets. Lastly, Loughran and McDonald (2016) have done a survey which aims to improve the understanding of textual analysis and its nuances.

This section is divided in two subsections. The aim is to provide the reader with an overview on the research done on two common topics within the field of textual analysis and finance. These topics are readability and sentiment.

2.2.1 Readability

To the extent of our knowledge, the academic discussion about textual analysis in combination with finance began with topics regarding readability and understanding of financial disclosure. Readability simply concerns determining what degree readers is able to understand the content of a written text. Smith and Smith (1971) found the readability on footnotes of Fortune 50 companies restrictive. Lebar (1982) compared information and topics between annual reports, 10-K's and press releases of 10 NYSE firms. Jones and Shoemaker (1994) concluded from their review on accounting readability that corporate annual reports are at a level of difficulty that makes it inaccessible to a large percentage of private shareholders. Jones and Shoemaker (1994) also looked into whether annual reports have been more difficult to read over time, finding it hard to conclude on the matter. Subramanian, Insley, and Blackwell (1993) discovered that it is significantly easier to read annual reports of profitable firms than of the ones that are performing poorly.

According to Loughran and McDonald (2016) a lot of earlier studies regarding readability should be taken with a grain of salt. They point out that most of the research on the topic, before Li's article in 2008, is done with to small samples or problematic

methodologies.

Li (2008) measures readability of annual reports by using the total sum of words and fog index.[4] He finds that annual reports with lower earnings are harder to read i.e. they are longer and with a higher Fog index. Bloomfield (2008) argues that this result may be due to poor performing firms feeling the need to use more sentences and text to explain their situation to the public.

Guay, Samuels, and Taylor (2016) finds that managers of companies with low readability in their annual reports tries to ease any negative investor reactions from this by conveying more forecasts of both sales, cash flows and earnings per share. They base this on six different readability measures whereby one is the fog index.

Loughran and McDonald (2014) finds that firms with larger 10-K file sizes are linked significantly to higher subsequent stock return volatility, analyst dispersion and absolute earnings surprises. They also reveal that the fog index is a poor measurement of readability in financial applications, and suggest using natural logarithms of 10-K file sizes as a proxy for readability.

Leuz and Schrand (2009) uncover that firms which increased the number of pages in their annual report after the Enron scandal to enhance their firm-specific transparency, was rewarded with lower cost of capital.

In the case of earnings calls, Loughran and McDonald (2016) suggests that measures which are more concerned about content is the right way to measure readability. The most extensive problem with respect to measuring readability in financial documents is to separate the document from the business. According to Leuz and Wysocki (2016),

[4]Fog index is a weighted sum of average sentence length in words and complex words. In this setting a complex word is a word with more than two syllables. Fog Index = 0.4(average number of words per sentence + percentage of complex words. (Li, 2008).

this is a fundamental problem that inflicts all accounting quality metrics.

2.2.2 Sentiment

Assessing the sentiment of text opens up for many applications when researching text and finance. Kearney and Liu (2014) defines textual sentiment as the degree of positivity or negativity in text. In the financial world, this is usually exploited to figure out the tone of financial disclosure. The tone can be used to decide if a text is more pessimistic or optimistic.

Deciding the sentiment of a text is usually done trough predefined word lists. These word list are often referred to as dictionaries. Dictionaries are settled on which words that relates to a positive sentiment and which words that relates to a negative sentiment. To exemplify, the word *terrible* contributes to a negative sentiment score, while *fantastic* would contribute to a positive sentiment score.

Textual sentiment might deliver insights within markets, firms, institutions and how they objectively reflect on their conditions. Textual sentiment can also provide some insight to investors subjective judgment and behavioral characteristics. An investors judgment and behaviour is typically more associated with the term *investor sentiment*.[5]

Tetlock (2007) exploited sentiment scores in his paper were he used a content analysis program called General Inquirer (GI). This program contains a dictionary called Harvard IV-4. Using this dictionary he found evidence that sentiment in the *Abrest the Market* column in Wall Street Journal could predict the movements of broad indicators on the stock market. This result is subject to criticism from Loughran and McDonald (2011).

[5]Baker and Wurgler (2007) defines investor sentiment as beliefs about future cash flows and investment risks that are not justified by observable information.

Loughran and McDonald (2011) demonstrated that the Harvard dictionary misclassifies words in the financial sector. They exemplify this using a wordplay in their paper title: *When Is a Liability Not a Liability*. In most cases, the word *liability* would be associated with a negative tone. However, for everyone dealing with accounting and the financial industry, *liability* is a common and neutral word. In financial literature, the most extensively used dictionaries are Harvard's GI, Loughran and McDonald (LM), Diction and Henry (Loughran and McDonald, 2016).

An aspect that might be relevant to consider when using text or words is a phenomenon called Zipf's law. Zipf's law states that word counts seems to follow a power law distribution. Manning and Schütze (2003) explains it "roughly" in the following way: a text will mainly consist of a few common words, then a moderate number of medium frequency words and a lot of low frequency words. The frequency of words tells how many times a word will appear in a text.

Loughran and McDonald (2016) provides an illustration of Zipf's law. They plotted the relative frequency for 10-K and 10-Q SEC filings between 1994-2012 which resulted in figure 4. This is a Zipf's curve. In this figure the Zip's curve can be found in both curves. One curve is based only on the negative words in the corpora. The other curve allows all words in the corpora to be included.

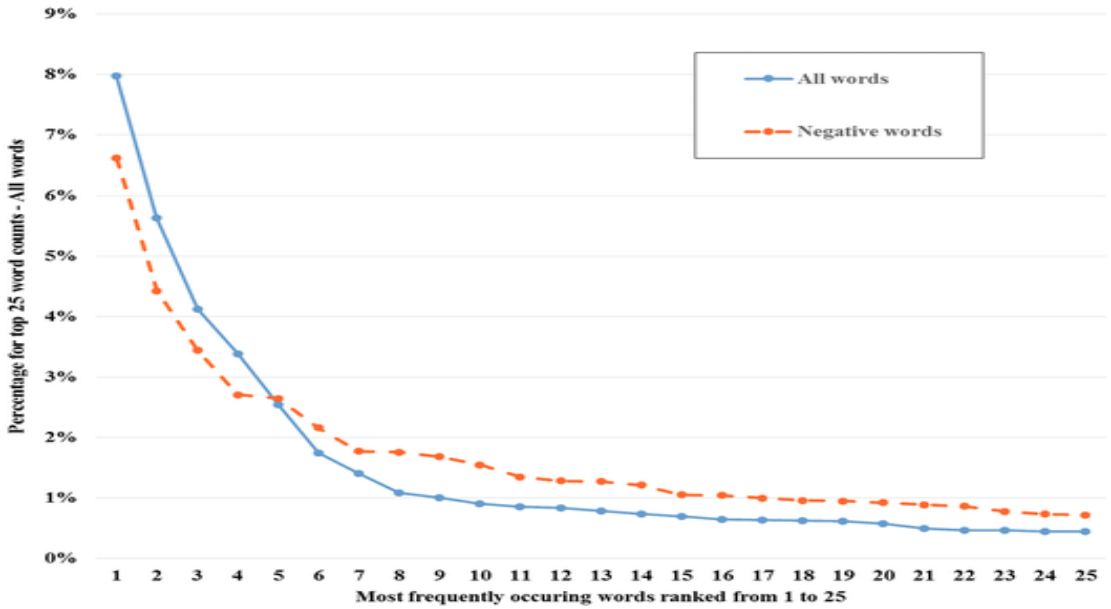


Figure 4: Example of Zipf's curve (Loughran and McDonald, 2016).

Figure 4 reveals how the top 25 words of the negative words list make up approximately 44% of the total text. The total negative word list consists of 2329 negative words. In other words, approximately 1% of the negative words makes up close to half of the document. This may be a source of error when trying to classify sentiment and tone. If words like liability and depreciation are driving the results, it is debatable whether a research paper can link a pessimistic undertone to their findings (Loughran and McDonald, 2011).

Another relevant problem when doing sentiment analysis is the aspect of context. A computational perspective will struggle to understand the context of what is written. Words with several meanings and concepts as irony is making analyzing tasks harder.

Reyes and Rosso (2011) tried in their paper to identify the key components for irony detection. This was done on ironic reviews collected from Amazon, with classifiers that

achieved decent levels of accuracy. Their best classifier was Support Vector Machine (SVM) and achieved an accuracy of 75,75%.

Kearney and Liu (2014) presents an extensive list on research done within textual sentiment on information sources as corporate disclosure, media articles and internet messages. This paper only include the ones that are relevant, i.e. using machine learning algorithms or earnings calls.

Henry (2006) finds that including predictor variables which capture verbal information and writing style, improves accuracy of market response predictions. He found this result using earnings press releases and classification and regression trees.

Price, Doran, and Peterson (2010) quantifies the linguistic tone of quarterly earnings conference calls for publicly traded real estate investment trusts. They find that the tone of the conference call dialogue has significant explanatory power for extraordinary returns at and immediately after an announcement. They also find that an overall positive tone between analysts and management in an earnings call discussion almost offset the disadvantageous effects of a negative earnings surprise.

Davis, Zhang, Ge, and Matsumoto (2015) takes a closer look at managerial tone in earnings conference calls. They find evidence which coincide with prior studies in that the market react to the overall tone of a conference call, as well as a manager specific element which impact investors. This result was found using the Henry and LM word lists.

Huang, Teoh, and Zhang (2014b) looked into whether firms deceive investors by using special forms of language in press releases. By using a large sample and the LM dictionary they find that an abnormal positive tone in earnings press release is significantly tied with low subsequent earnings and cash flows. This effect can last as long as three

years after the initial release.

Twedt and Rees (2012) examined whether tone and detail are significant in markets and their response to analysts' reports. Using the LM dictionary to measure tone, they find that the tone of financial analyst reports include significant information, which adds value to a reports' earnings forecast recommendations.

In combination with using word counts there are a considerable literature regarding how computational logistics should be normalized.[6] In most cases the raw word count is of no interest since this is strongly tied with the length of the document. An easy way to solve this is the use of proportions.

Adjusting the weight a term receives based on how unusual it is is very useful. Usually the same word will be used throughout a document. In Loughran and McDonald (2016) they state that the word *unfavourable* appears a 1000 times more often than *misinform* or *expropriating*. This suggests that the use of *misinform* or *expropriating* might relate to something more serious.

In her paper, Jones (1972) comes up with a way to assign more weight to unusual terms. This is done through the *term frequency - inverse document frequency* (tf-idf). Tf-idf is a way of normalizing textual data and is widely used within computational text analysis.

2.2.3 Correspondence Analysis and Cosine Similarity

A major part of textual analysis is to automate recognition of similar documents. This is useful if the researcher wants to sort unclassified and independent documents into their most likely group based on how alike they are. Two well known methods for visualizing

[6]For more information see Salton and Buckley (1988) or Zobel and Moffat (1998).

document similarities are *correspondence analysis* and *cosine similarity*.

Correspondence analysis is a multivariate analysis technique for exploring cross-tabular data by converting such tables into graphical displays called "maps," and related numerical statistics (Greenacre and Blasius, 1994). This analysis reveals the structure of a complex data matrix without losing essential information by mapping associations between rows and columns in a frequency table, which makes it possible to plot the points in a space of few dimensions (Clausen, 1998). The method had its first mathematically application by Hirschfeld (1935), and was rediscovered much later in France in the 1960s and has since then largely been used for graphical data presentation (Greenacre and Hastie, 1987).

The main essence is that correspondence analysis will show associated column- and/or row profiles plotted together in two-or three dimensions, but because of vast reduction of dimensions, one can only look at how they cluster, and not interpret the relative distances.

Another way to measure how similar documents are is the cosine similarity measurement. When documents are represented as term vectors, the similarity of two documents can be measured as the correlation between their corresponding vectors. The cosine similarity is then further quantified as the cosine of the angle between the two vectors (Huang, 2008). Cosine similarity is easy to compute and is useful for comparisons because it is defined to be between 0 and 1.

The cosine similarity between two documents $\overrightarrow{t_a}$ and $\overrightarrow{t_b}$ is:

$$SIM_c(\overrightarrow{t_a}, \overrightarrow{t_b}) = \frac{(\overrightarrow{t_a})\,(\overrightarrow{t_b})}{|\overrightarrow{t_a}|\,x\,|\overrightarrow{t_b}|}, \tag{2.1}$$

where $\overrightarrow{t_a}$ and $\overrightarrow{t_b}$ are m-dimensional vectors over the term set $T = \{t_1, \ldots, t_m\}$. Each dimension represents a term with its weight in the document, which is non-negative (Huang, 2008).

Cosine similarity has been widely applied in textual analysis. For example, Hoberg and Phillips (2016) analyzed companies' 10-K product descriptions to find out how firms differ from their competitors. Based on the cosine similarity, they determined that firms can have their own set of distinct competitors which is different from the explicitly outspoken ones.

2.3 Machine Learning Algorithms for Classification

Both human and machine intelligence relies heavily on classification. Classification is the action of putting ideas and objects into a category, through a process of recognition, differentiation and understanding.[7] Determining what image, letter or word has been exposed to our senses, sorting mail, or recognizing voices and faces are all examples of assigning a label to an input. Many tasks within textual processing are classification tasks. Some examples are rating customer reviews as positive or negative, sort out spam, decide author attributes or simply putting a label on a text.

In order to attain some intuition on how textual classification greatly simplifies both business and personal life, consider how an email service manage to sort out fraud attempts. This is done by teaching a machine learning algorithm to label any receiving text as junk if it meets a set of criteria e.g. an email where the subject field has only capital letters and the body contains an extensive use of exclamation marks. Considering the amount of emails sent and received every day, and the variety of labels an email can

[7] Our definition.

be assigned, the utility of not needing to do this manually is obvious.

This section will present the classification techniques used in this thesis. These machine learning algorithms are naive Bayes, logistic regression with lasso regularization, stochastic gradient boosting and SVM with linear kernel.

2.3.1 Naive Bayes

Multinomial naive Bayes is a probabilistic classifier. It is based on Bayes theorem, which is credited Bayes, for an essay he wrote in 1763.[8] Khan, Baharudin, Lee, and Khan (2010) claims that naive Bayes is a widely studied and a popular go to algorithm within text classification.

The name comes from the simplifying (naive) assumptions about how the features interact. With natural language processing in mind, the first simplifying assumption is that a words' placement in a text is irrelevant. This means that words has the same effect whether they occur as the 1st, 35th or last word in a document (McCallum and Nigam, 1998).

The second assumption is frequently called the *naive Bayes assumption*. This condition assume that the value of a particular feature is independent of the value of any other feature, given the class variable (McCallum and Nigam, 1998). For this reason, a feature's value and probability can be *naively* multiplied with other values and probabilities, given same class. In the end, naive Bayes will classify based on which combination of features that returns the highest probability, given same class, thus earning the title *probabilistic classifier*.

Maron (1961) was the first to suggest using multinomial naive Bayes for text classifi-

[8] See: *An Essay Toward Solving a Problem in the Doctrine of Chances*.

cation. In the same year, Minsky (1961) proposed the naive Bayes classifier to solve artificial intelligence problems. Mosteller and Wallace (1963) was the first to apply Bayesian analysis on a classification problem within text. They used this to decide the authorship of 12 essays from the Federalist Papers.[9]

Antweiler and Frank (2004) was the first to use naive Bayes on a financial topic. They studied messages posted on *Yahoo! Finance* and *Raging Bull*. They found that message boards reflect the views of day traders and that more disagreements on the posting sites is followed by an increased trading volume.

Li (2010) uses the naive Bayesian machine learning algorithm to examine information content of the forward-looking statements (FLS) in the management discussion and analysis (MD&A) part of 10-K and 10-Q filings. He finds that the average tone of the FLS is positively associated with future earnings. When discussing future operations, a more positive tone is associated with higher future earnings for the firm.

Huang, Zang, and Zheng (2014a) uses a naive Bayes machine learning approach to deal with the sentiment in 363,952 analysts reports. They find that find that investors react more strongly to negative than to positive text, suggesting that analysts are especially important in propagating bad news.

In their paper, Buehlmaier and Whited (2014) uses naive Bayes to predict the probability of a firm being financially constrained using MD&A text from 10-K filings. They find that higher stock returns are associated with firms that are more financially constrained.

Buehlmaier and Zechner (2013) proves that information about sentiment within news

[9]The Federalist Papers is a collection of 85 articles which aims to promote the ratification of the United States Constitution.

media stories only slowly incorporates into stock market valuations. They illustrate this using naive Bayes methodology to measure sentiment on newspaper articles regarding merger announcements in the US.

2.3.2 Lasso

Lasso or *Least Absolute Shrinkage and Selection Operator* was introduced by Tibshirani (1996). It is a logistic regression method that tries to improve prediction accuracy and interpretability of a statistical model. This method achieves improved prediction accuracy through regularization and variable selection.

Tibshirani (1996) defines the lasso estimate in the following way:

$$\sum_{i=1}^{n}\left(y_i-\beta_0-\sum_{j=1}^{p}\beta_j x_{ij}\right)^2+\lambda\sum_{i=j}^{p}|\beta_j| \tag{2.2}$$

Here x_i are predictor variables and y_i are responses. The equation is a trade off between two different criteria. Lasso regression seeks coefficient estimates that fit the data well by making the sum of squared residuals small through the first part of the expression. However, the expression also comes with a shrinkage penalty, which is the second part of the expression. λ takes the role as a tuning parameter, controlling the impact of the two terms in the expression. λ is determined separately and selecting the correct value is critical.

By forcing the sum of the absolute value of the regression coefficients to be less than a fixed value, Lasso makes certain coefficients is set to zero. This effectively creates a model that is simpler and which does not include zero set coefficients. This way the

residual sum of squares is minimized and the model is more interpretable. Since a lasso model only will keep a subset of the variables, it is common to say that lasso yields *sparse* models.

In their paper, Skianis, Rousseau, and Vazirgiannis (2016) presents regularization and shrinkage methods as important techniques in language processing and classification tasks. It is reasonable to conclude that these methods are especially advantageous in eliminating the large amount of noise contained in textual data when trying to detect signals for text classification tasks.

2.3.3 Stochastic Gradient Boosting

Gradient boosting is an ensemble method. Ensemble methods solves predicting problems by using a collection of predictors, which together provides a final prediction. The advantage of ensemble methods is the use of many different predictors, which together will do a better job than any single predictor alone.

Gradient boosting employs the logic in which the subsequent predictors learn from the mistakes of the previous predictors (Friedman, Hastie, and Tibshirani, 2000). Therefore, the observations have an unequal probability of appearing in subsequent models and the ones with the highest error appear most. The predictors can be chosen from a range of models like decision trees, regressors, classifiers etc. Because new predictors are learning from mistakes committed by previous predictors, it takes less iterations to reach close to actual predictions.

Natekin and Knoll (2013) highlights that gradient boosting is particularly efficient at dealing with a large number of features. It also excels by being a suitable way to reduce bias. The drawback is that this makes gradient boosting prone to overfitting.

Friedman (2002) modified gradient boosting, resulting in the term Stochastic Gradient Boosting. Specifically, he proposed that at each iteration of the algorithm, a base learner should be fit on a subsample of the training set drawn at random without replacement. With this modification he got a substantial improvement in gradient boosting's accuracy.

2.3.4 Support Vector Machine with Linear Kernel

Support Vector Machine is a supervised machine learning algorithm which can be used for both classification or regression problems (Cortes and Vapnik, 1995). In this algorithm, each data item are plotted as a point in n-dimensional space. N is the number of features, with the value of each feature being the value of a particular coordinate.

SVM does classification by finding the hyperplane that differentiate the two classes best.[10] There are many hyperplanes that might classify the data. One reasonable choice as the best hyperplane is the one that represents the largest separation, or margin, between the two classes.

The behaviour of SVM can be changed by the use of Kernels. This makes is possible for SVM to create non-linear hyperplanes. This thesis use linear kernel. The rational behind this is due to previous research findings. Firstly, classification problems are usually linearly separable (Thorsten, 1998). Secondly, according to Hsu, Chang, and Lin (2016), linear kernels is faster and works well when there are a lot of features involved. On this premise they establish that mapping the data to a higher dimensional space does not improve the performance. This makes linear kernel a suitable alternative due to reduced computational cost.

[10]Hyperplane is a subspace whose dimension is one less than that of its ambient space. If a space is 3-dimensional then its hyperplanes are the 2-dimensional planes.

2.4 Machine Learning Concepts

This section will go through two important concepts and distinctions regarding machine learning. These concepts are over- and underfitting, and supervised and unsupervised learning.

2.4.1 Overfitting and Underfitting

Overfitting and underfitting are important concepts within machine learning. Tušar, Gantar, Koblard, Ženko, and Filipiča (2017) defines overfitting as the result of an algorithm that is to customized to the data set and picks up noise instead of underlying relationships. This leads to a prediction algorithm that may fail to fit additional data or predict future observations reliably. Overfitting is often the result of a to complex model i.e., extensive use of features.

Aalst, Rubin, Verbeek, Dongen, Kindler, and Günther (2010) explains underfitting as the result of a model that fails to learn the underlying relationship in a data set. This way the model generalizes to much and describes a model that is too simple with regards to the data it is trying to model. To overcome problems with underfitting it is recommended to increase the complexity of a model.

Overfitting or underfitting can be detected trough assessing the bias and variance of a model (German, Bienenstock, and Doursat, 1992). A model which overfits has high variance and low bias on the training data. This leads to poor generalization on testing data. A model which underfits has low variance and high bias on the training data. Generalization on unseen data may here seem adequate, but the model will have very poor prediction performance.

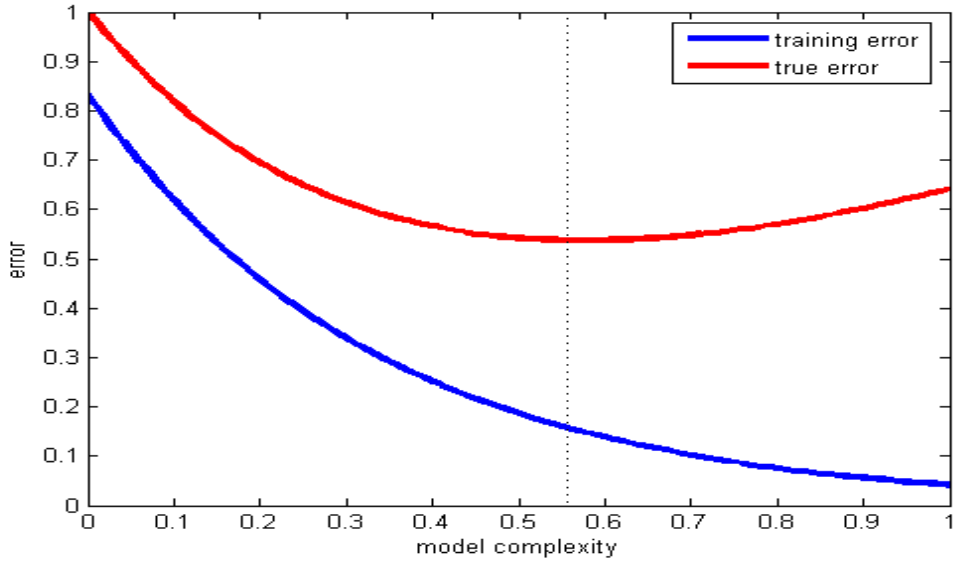


Figure 5: Common model fitting pattern (Pennsylvania, n.d.)

Figure 5 is an graph, displaying the case of overfitting and underfitting. The red line illustrates the true error of this classification problem, while the blue line illustrates a model trained on the same data. The dotted line illustrated the optimal complexity for a model, approximately 0,55.

The optimal solution for this problem is to develop a model that is identical with the red line. Unfortunately, the true error of a classification problem is rarely observable. The point where the red line and the dotted line crosses displays the optimal trade-off between minimizing training error and model complexity. Prior to this point, the red line is underfitting and after this point the red line overfits.

The blue line illustrates a common pattern of a model fitting to training data. Prior to the dotted line, the error is high due to a model not picking up underlying relationships. As the complexity of the model increase, the error of the model is reduced. High val-

ues of complexity reduced the error of the model sharply. This might cause a belief of a well performing model, but the model is now overfitting, making it insufficient on doing predictions. This is why machine learning models are subject to different validation techniques. This thesis provides theory around validation techniques in section 2.5.1.

2.4.2 Supervised and Unsupervised Learning

Machine learning algorithms can be divided in two main groups. These groups are supervised machine learning and unsupervised machine learning. The names refer to how the algorithms learns.

Supervised learning is algorithms were the data scientist acts as a guide to teach the algorithm what conclusions it should come up with. A comparison could be the way a child might learn arithmetic from a teacher. Supervised learning requires that the algorithm's possible outputs are known and that the data used to train the algorithm is already labeled with correct answers (Hastie, Tibshirani, and Friedman, 2009).

Supervised machine learning algorithms makes it possible to solve classification and regression problems (Hastie et al., 2009). Classification is used to identify where a data point belongs, given a set of categories. Regression is used to predict a continuous value. To exemplify, classification could be used to predict whether a stock price is moving up or down, while regression would be used to predict the actual value of a stock.

On the other hand, unsupervised machine learning follows the idea that a computer can learn to identify complex processes and patterns without a human to provide guidance along the way (Hastie et al., 2009). Although unsupervised learning might be more complex and difficult to understand, it opens the doors to solve problems that are very

difficult for humans to conquer.

Unsupervised machine learning makes it possible to extract hidden structures in a data set (Hastie et al., 2009). Some ways to achieve this is trough clustering or association. Clustering is used to group observations based on similarity. Association is used to detect rules that describes large portions of a data set. Given a data set based on NYSE stocks, clustering might be able to identify which companies that operate within the same industry, while association could discover which companies are negatively correlated.

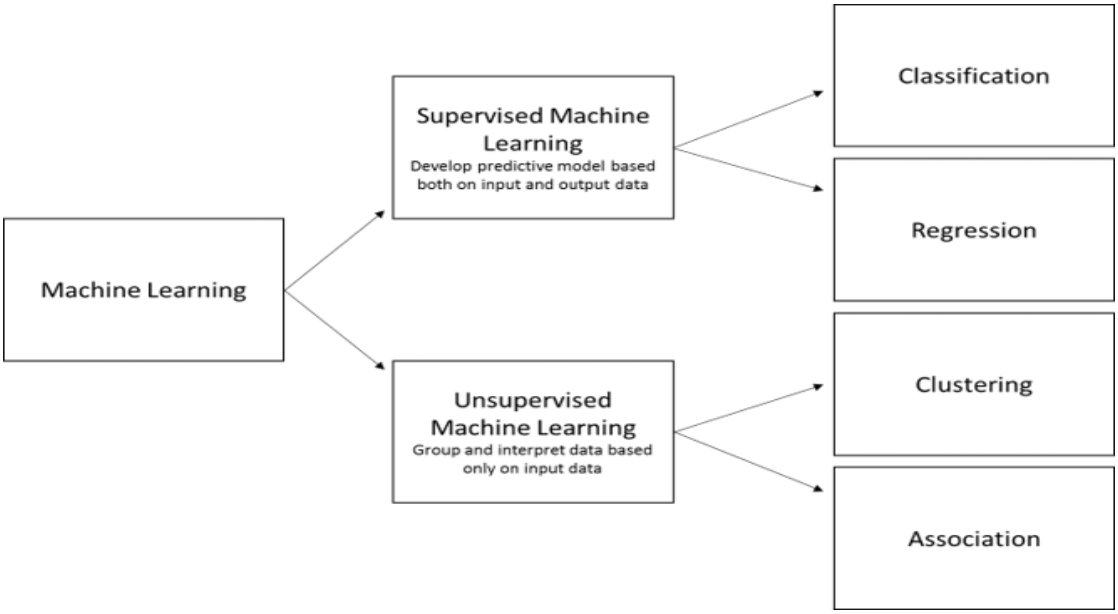


Figure 6: Difference between supervised and unsupervised learning

Figure 6 displays how machine learning algorithms is grouped. Choosing to use either supervised or unsupervised machine learning algorithms usually depends on factors related to the volume and structure of the data and the objective. In some cases it might be beneficial to use both types of algorithms to build predictive data models.

2.5 Performance Measures

To assess the performance of the prediction models this thesis make use of a test and validation technique, accuracy and error measure. The following two subsections will present theories on these performance measures.

2.5.1 Testing and K-Fold Cross-Validation

In a prediction problem, it is usual to divide a data set in three parts. These parts are called training set, validation set and test set. Usually a large part of the data is portioned to the training set and the rest divided on the validation set and test set e.g., 60%, 20% and 20%. This structure is used to see how a model generalizes to independent data sets and avoid overfitting (Tušar et al., 2017).

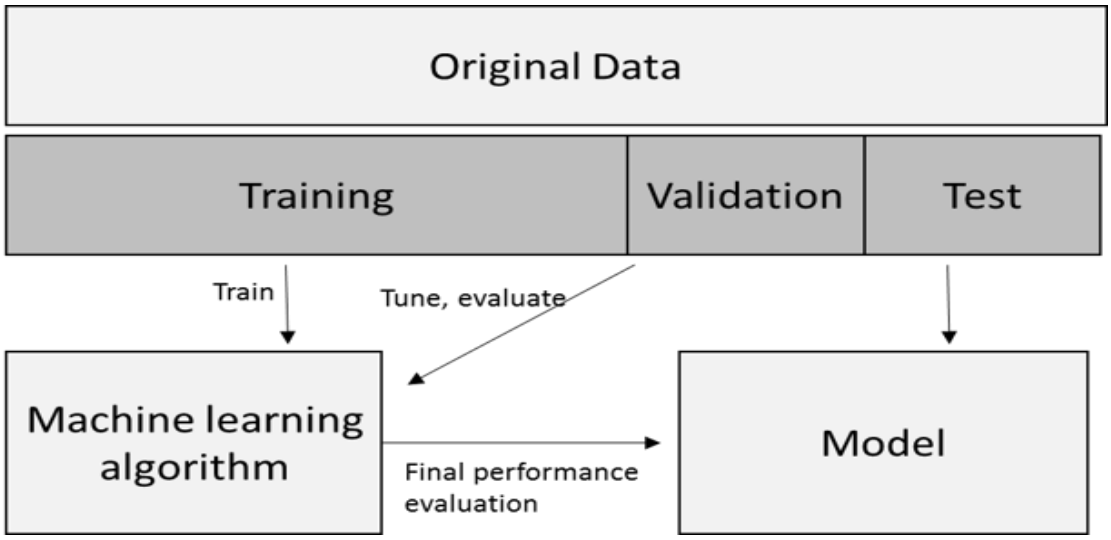


Figure 7: Process of generalizing results of prediction problems

Figure 7 illustrates the process of creating a supervised learning model and see how well it generalizes. The training set is used to train the model, i.e. an algorithm utilize

the training set and find distinct relationships between the predictors and the response variable. Then the model is validated. At this stage the researcher use the model to do predictions on a new set of observations whereby the aim is to tune parameters of the algorithm to improve the predictions.

When finished with the training and validation part, the tuned model which is believed to perform better is tested. The model predicts outcomes on the test subset. The test set is new, unseen and untouched observations. The researcher will then assess how well the model performs. If the performance drops significantly, it can be concluded that the model is subject to overfitting and does not generalize well.

K-fold cross validation is a model validation technique that builds on the same principles as above, but combines the training set and validation set. This makes it possible for a machine learning algorithm to train on more observations (Arlot and Celisse, 2010). The training and validation set is now divided in K-subsamples, were one subsample is the test set and the remaining subsamples are the training set. This process is continued K times, so that all observations is part of the test subsample once, and $K-1$ times as the training subsample.

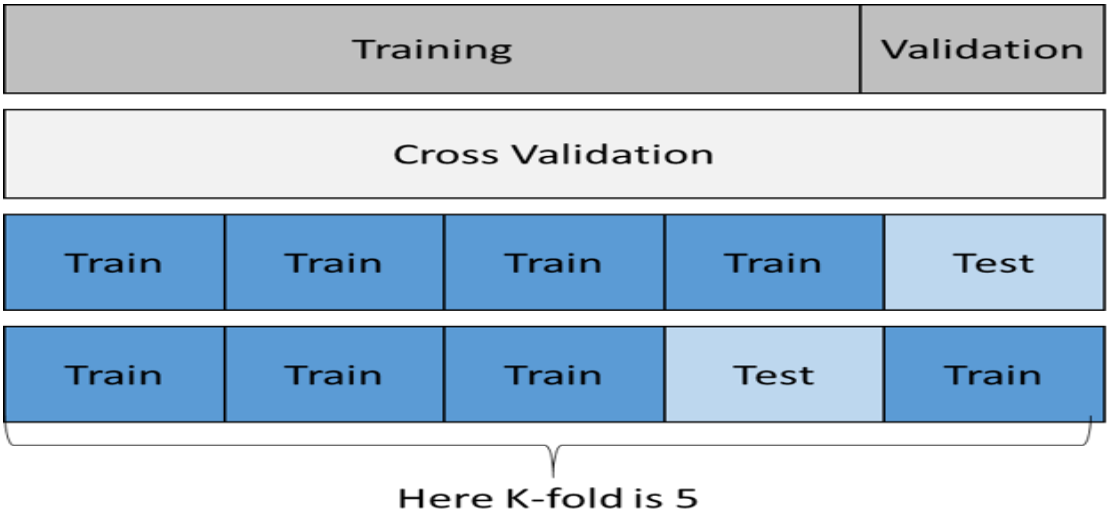


Figure 8: Process of cross validation.

Figure 8 illustrates how the cross validation is done where $K = 5$. After cross validation is finished, the researcher has to decide on which algorithms to proceed with to the test set.

2.5.2 Accuracy and Error

Evaluating how well a model performs are usually done through evaluating the accuracy or error of a model. Figure 9 is provided to understand these metrics.

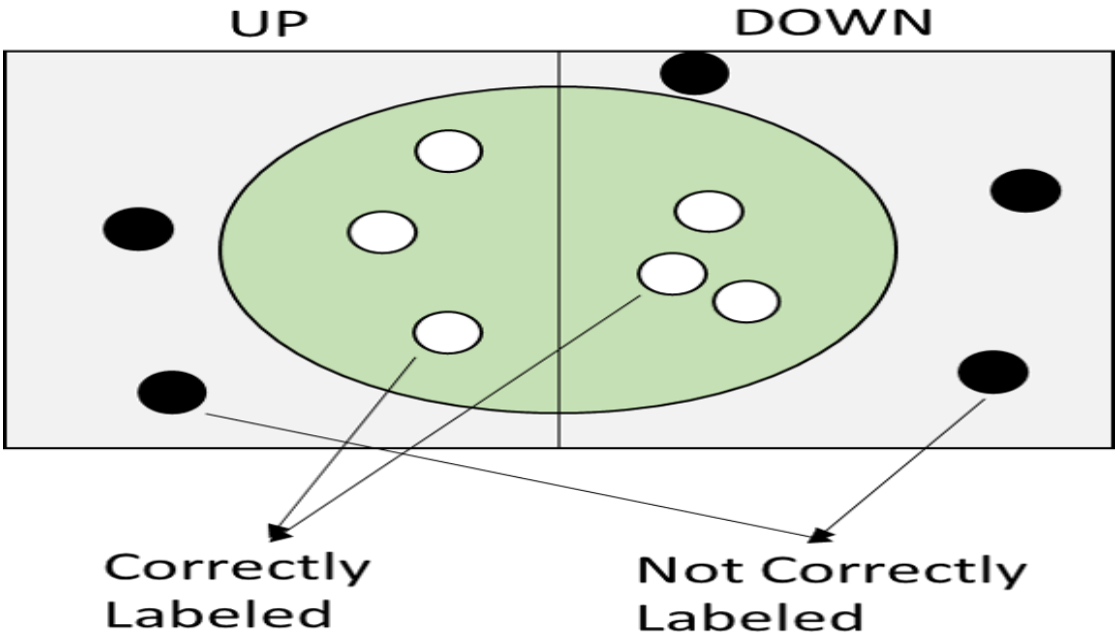


Figure 9: Accuracy and error

Figure 9 illustrates that a high accuracy score is achieved through maximizing correctly labeled observations. This means that transcripts that are labeled up, with the corresponding stock price going up after an earnings conference call, are correctly labeled. The same principle applies for observations which are labeled down, with a corresponding stock price that went down after the earnings conference call. Every other combination are not correctly labeled and will reduce the accuracy of the model. This is known as error.

The following formula is used to calculate the accuracy of the model:

$$Accuracy = \frac{correctly\ labeled}{total\ amount\ of\ observations} \tag{2.3}$$

Accuracy and error mirror each other as metrics in that error is calculated by $1 -$

accuracy. Machine learning programs often return one of these values as a default.

In many cases, accuracy is a natural metric, but it struggles when dealing with unbalanced classes (Metz, 1978). If the researcher in addition to having unbalanced data, also aim to solve a classification problem where misclassification costs are not the same for false positives and false negatives, other performance metrics should be used. This can be illustrated through a spam detector example. Imagine a set of 1000 emails where 50 of them is spam. If a classifier labels every email as non-spam, this classifier would achieve an accuracy of 95% and error of 5%. However, the classifier is unsuitable since it did not detect any spam emails. Section 4.1 unveils whether or not the data set used in this thesis is balanced.

3 Methodology

This section will explain the approach and the choices that are made to answer the research question. In particular, this section will take a closer look at the research question, response variables, performance measures and the data.

3.1 Research Question

The aim of this thesis is to answer the following research question:

How much predicting power is attainable through textual analysis of earnings call transcripts to correctly predict stock price movement the next day.

This thesis tries to answer the research question through a combination of financial theory, text mining and machine learning algorithms. Why this is a suitable approach will be presented and discussed in the following.

Firstly, there is valuable information in the transcripts of an earnings conference call. Previous research confirms this and it makes sense from an economic standpoint. Stock performance and future performance is heavily dependent on the amount of money a company generates. An earnings call offers the possibility to inspect the revenue stream, potential issues and how a company plans to deal with those issues.

Secondly, an earnings call offers the possibility to assess the leaders of a given company and the potential quality of them. The calls can provide insight into the leadership and their knowledge, confidence, interaction, etc. Lastly, the Q&A session is a potential source of incremental information regarding companies' results and numbers.

Earnings conference calls transcripts generates a lot of text, and to achieve the goal of

this thesis the text needs to be organized. Different text mining techniques is used to create a web crawler[11] to download transcripts, create the data set and reduce the amount of noise within the data set.

Reducing noise is done through stemming, and through removal of both stopwords and words without information. Reducing the noise has the benefits of reducing computational time and complexity, making it less challenging for machine learning algorithms to detect relations in the data set. This might increase the accuracy of the prediction models and are explained in more detail later.

This thesis use the programming language R for all tasks i.e., data collection, preprocessing and predictions. The reason for choosing R comes down to familiarity with the program language and a relatively large online support community.

We focus on four well documented supervised learning algorithms that are fast and easy to implement using packages from R. These are naive Bayes, logistic regression with lasso regularization, stochastic gradient boosting and support vector machine. Initially, our aim was to include an extensive and far-reaching list of learning algorithms. However, because of immense amounts of data, the computations becomes expensive and time consuming.[12] It is assumed that the results of the models used in this thesis gives a good picture of the predicting power of earnings call transcripts.

Another solution to restrictions on computing power is to reduce the data. This might be eligible in this thesis, but as long as we do not underfit, and the model incurs no high bias, more observations generally will generalize better on unseen data. Thus this thesis moves forward by maximizing the amount of data.

[11] A program that automatically copies and downloads text from websites

[12] Smaller subsets of data were used to proxy full-data performance for other learning algorithms, but did not yield any clear signs of improvement compared to the included learners.

The thesis investigates transcripts from NYSE and NASDAQ only. This choice is made to benefit on the fact that all companies on these stock exchanges are having earnings conference calls. This makes it simpler to collect a sufficient amount of observations.

The time period 2014-2017 is chosen to work with recent data. This avoids trouble in dealing with financial crisis periods and potential problems with an unbalanced data set.

3.2 Response and Features

Depending on different factors, all firms will hold their earnings conference calls on different times. The following two subsections will explain how this is dealt with.

3.2.1 Response

To train a classifier that predicts whether a stock price is going up or down the day after a call is held, the response variable is the next day's return of the stock. Due to the fact that many earnings calls are held outside of the time interval 09.30 to 16.00, which is the trading hours of NYSE and NASDAQ, the computation of responses becomes more challenging. How the response variable is derived is dependent on what time of the day the earnings call is held.

Table 3 reveals the distribution of starting points of the calls. The different times are stated in eastern time. Many calls are conducted outside of the trading hours, and some at night time. The explanation for the more uncommon time periods is that a great share of companies in the data set are foreign and consequently some are holding their calls in other time zones.

Table 3: Frequency and time of calls

Time	N	G
24:00 to 3:00	102	0
3:00 to 6:00	259	0
6:00 to 9:30	9,180	0
9:30 to 12:00	9,467	1
12:00 to 16:00	1,724	1
16:00 to 18:00	8,789	2
18:00 to 21:00	274	2
21:00 to 24:00	65	2

We are interested in recording market reactions following a conference call. As seen in table 3, the calls are set apart in three groups. One group for calls held before trading hours (G_0), one for the calls that are held in the trading hours (G_1), and one for calls held after trading hours (G_2). The three different groups have different ways of deriving the response variable. This is done to obtain an accurate depiction of market movements following a conference call.

The response variable is derived in the following way:

$$\frac{CLOSE_{i,t}}{CLOSE_{i,t-1}} - 1 = \begin{cases} Y_{i,t+1} = 1 & if\ > 0 \\ Y_{i,t+1} = 0 & if\ < 0 \end{cases}, for\ G_0$$

$$\frac{CLOSE_{i,t+1}}{OPEN_{i,t}} - 1 = \begin{cases} Y_{i,t+1} = 1 & if\ > 0 \\ Y_{i,t+1} = 0 & if\ < 0 \end{cases}, for\ G_1$$

$$\frac{CLOSE_{i,t+1}}{CLOSE_{i,t}} - 1 = \begin{cases} Y_{i,t+1} = 1 & if\ > 0 \\ Y_{i,t+1} = 0 & if\ < 0 \end{cases}, for\ G_2$$

Where *i* is the individual stock, *t* is the date at which the earnings call transcript is held, and CLOSE and OPEN is the closing and opening price of the stock where it is trading (NYSE or NASDAQ).

3.2.2 Features

This master thesis uses the bag of words method (see section 2.1). Assessing what predictors, in this case terms, to include in the model involves certain different steps. The data is partitioned randomly for the train and test sets with respectively 80% and 20% of the documents. Following this, the two sets are made into separate document term matrices. This results in two matrices, the train matrix with 23 471 rows, and the test matrix with 5868. Both matrices has roughly 3000 columns which corresponds to the unique terms of the whole corpus after cleaning.[13]

Each point in the matrix is the frequency of a word in a given transcript. Thus, the

[13] See section 3.5 for the cleaning steps.

matrix ends up being very sparse.[14]

The terms that only appears in just a few documents is of little interest and creates a lot of noise. Thus, the terms that appear in less than 5% of the documents are removed. This method of feature selection for text mining is both inexpensive and efficient. This shrinks the total amount of different words down to around 3000. The terms that are present in the training set but does not occur in the test set are also erased. This way, the model only predict outcomes with features at which it has learned. After creating the document term matrices and the initial processing, term frequencies are normalized into *Term frequency - inverse document frequency*.

Term frequency - inverse document frequency

This thesis follows Loughran and McDonald (2011) which uses one of the most common tf-idf techniques in their paper. They also adjust it to account for document length. This results in the following equation where df_t is the number of documents in a group of documents including the term t. N is the total amount of documents.

$$idf_t = log\frac{N}{df_t} \tag{3.1}$$

if $tf_{t,d}$ is the raw count of term t in document d, and a_d is the average word count in document d, then:

$$tf - idf_{t,d} = \frac{1+log(tf_{t,d})}{1+log(a_d)} log\frac{N}{df_t} \qquad if \quad tf_{t,d} \geq 1 \quad otherwise \quad 0 \tag{3.2}$$

Term frequency $tf_{i,j}$ counts the number of occurrences $n_{i,j}$ of a term t_i in a document

[14] A matrix in which most of the elements are zero.

d_j. *Inverse document frequency* for a term t_i is defined as

$$idf_i = log_2 \frac{|D|}{|\{d|t_i \in d\}|} \tag{3.3}$$

where $|D|$ denotes the totalt number of documents and where $|\{d|t_i \in d\}|$ is the number of documents where the term t_i appears. *Term frequency - inverse document frequency* is now defined as $tf_{i,j} \times idf_i$.

3.3 Performance measures

To assess the ability to generalize the results and performance of the predictive models, cross validation is used. Also, a benchmark is generated to compare the predictive power of the models. The following will present the choices that are made and the reasoning behind this.

3.3.1 5-fold Cross Validation

With aim of generating models that generalizes to unseen data, 5-fold cross validation is applied. 80% of the total data set are randomly assigned to cross validation and the remaining 20% to testing. Cross validation is chosen to take advantage of training and validating at as many observations as possible. Critics might suggest that better performance can be achieved by 10-fold cross validation, as Borra and Di Ciaccio (2010) demonstrate in their paper. This thesis uses 5-fold cross validation to benefit on reduced computational cost.

3.3.2 Accuracy Benchmark

To evaluate how well the models performs, a benchmark is constructed. The benchmark is constructed using the same supervised learning algorithms used for the textual classification models. For the benchmark, the only feature included to predict the stock direction is the previous day's S&P500 index. Thus, the benchmark is generated through available market data at the same time the researcher also would have access to the earnings call.

Following Stapor (2017) and Nadeau and Bengio (2003), we conduct a *corrected resampled t tests* which is used to compare the text classifiers and the classifiers utilizing S&P500 data. This is an repeated estimation method in i-th of the m iterations, where random data partition is conducted and the classification accuracies for test data $A_{k1}^{(i)}$ and $A_{k2}^{(i)}$ of compared classifiers *k1* and *k2*, are obtained. The t-statistic is:

$$t = \frac{\overline{A}}{\sqrt{(\frac{1}{m} + \frac{N_{test}}{N_{train}}) \sum_{i=1}^{m} \frac{(A^{(i)} - \overline{A})^2}{m-1}}} \tag{3.4}$$

Where $\overline{A} = \frac{1}{m}\sum_{i=1}^{m} A^{(i)}$, $A^{(i)} = (A_{k1}^{(i)} - A_{k2}^{(i)})$, and N_{test}, N_{train} are the number of samples in the partitioning sets.

The benchmark is constructed with the aim of generating models that recognize responses in tomorrows stock prices based on movements of the broad equity market. Thus, the benchmark is mainly introduced to provide a more intuitive understanding of the performance for readers and for analytic purposes. Also, due to lack of general guidelines regarding best practice benchmark with respect to the classification task, the choice appear suitable.

3.4 Data Collection

To investigate the predictive power of earnings calls, it is needed to collect transcripts from earnings calls. Using machine learning algorithms to see if these algorithms will detect hidden value, creates a need for large amounts of textual data. The initial aim was to download and create a data set consisting of around 80 000 earnings call transcripts. Another reason for setting the aim high was due to a suspicion of loosing observations to different errors and cleaning purposes.

The data was collected utilizing a web crawler coded in R. The webcrawler crawled the website seekingalpha from page 1 to 2400.[15] This resulted in earnings call transcripts from December 2017 back to January 2014. The collecting process were time consuming and took over a month to complete.

29 339 earnings call transcripts was collected and saved. The transcripts are from 3689 unique firms, whereby 1745 are NASDAQ-firms and 1944 are NYSE-firm.

The crawler was coded to populate a data frame of four columns; the downloaded texts, the date of the call, the time of day the call was held, and the respective ticker of the stock. The tickers where used to identify companies on *The Center for Research in Security Prices (CRSP)* and to obtain the accordant stock prices.[16] The stock prices were then matched with the data set using tickers and dates. This resulted in a data set which contained firm name, ticker, date, earnings call text and stock prices of the day of the earnings call and the subsequent day.

The stock prices made it possible to identify the movement of the stock prices the day after the earnings conference call. This was used to create a new column which classi-

[15] The crawler was designed to mimic a human user to avoid overloading seekingalpha's servers and disrupt other users experience.

[16] Access to CRSP was gained through NHH and Wharton research data services.

fied the movement *UP* or *DOWN* on each row.[17]

3.5 Data Preparation

Working with textual data can be challenging, especially if the end goal is to build an high performing classifier. With almost 30 000 transcribed texts of oral origins, there are multiple aspects that could compromise the data and results.

Firstly, in addition to words and sentences that carry no real information in regards to the classification objective, there is an enormous amount of noise in the text since the web crawler extracts the source code of the websites. Section 3.5.1 explains how this noise was minimized.

Secondly, the oral original format of the text means frequent use of common words for human face-to-face interaction. These words and statements are driven by the nature of an earnings call transcript. To clean the texts of this, stop words and uninformative words are removed.[18]

Lastly, a bias can occur as the text have a large variance of words and sentence structures which essentially has the same meaning, but has entirely different linguistic architecture. Ideally, this thesis wants a data set that do not differentiate between this. When using the bag of words method, one way to combat this is to stem the whole corpus.[19]

[17] See section 3.2
[18] See section 3.5.2
[19] See section 3.5.3

3.5.1 General Text Cleaning Procedures

After retrieving raw text from the websites, the next step is to clean the text. How the cleaning will be done is determined by the goal of the text miner. In general however, if the goal is information retrieval in its broad sense, there are some common approaches and normative moves when cleaning text.

The raw text will typically contain HTML-tags, non ASCII letters, excess white space, end of sentence characters (punctuation characters) and numbers. A common way to extract a clean text is to remove all of the above. In this thesis this was done for the raw text. All charachters in the corpus is also transformed to lower-case characters. In addition, all contractions are changed e.g. ”won’t” or ”´ve” etc. to its extended form, namely ”will not” and ”have”.

3.5.2 Removal of Stop Words and Uninformative Words

Stop words are frequently used words of the English language, and removal of stop words is one of the more common steps in natural language processing. The idea is to remove words that occur frequently across all documents of the corpus. These words have absolutely no significance to the objective of the researcher, and serve only as noise. Articles and pronouns are the most typical stop words. (Hardeniya, Perkins, Chopra, Joshi, and Mathur, 2016)

There are multiple ways to discard these words. For example, the words that occur frequently and across all documents can be found. This way, the text miner would extract a list of uninformative terms that is specifically designed for the corpus on his hands. In this thesis, a short list of manually selected words was used to exclude terms occurring often due to the oral format of the transcripts e.g ”thank you”, ”good morning”, ”wel-

come", "quarter" etc.

A useful and very common way to reduce noise from general stop words is to load a predefined dictionary of well known stop words and remove all the words that occur in that list from the corpus. Such a list is the "SMART Information Retrieval System" stop words from Cornell University which is the one used in this thesis.[20]

3.5.3 Stemming

Stemming is the process of using a stemming algorithm to reduce all words in the text into their respective *stems*. An example of the process would be changing the words "computing", "computer", and "computational" all into "comput". Since the aim of bag-of-words models is to collect frequencies of unique words, stemming greatly reduce the feature dimension of the document term matrix.

This thesis uses the well known "tm" package from the Comprehensive R Archive Network (CRAN) to stem the texts. However, all stemming algorithms will make mistakes when operating on words without any additional grammatical information such as part-of-speech tagging.[21] An example is the sentence "the costumers seemed bored by the new product". With respect to the word "bored", the stemmed result is ambiguous. It is questionable whether a specific stemming method will treat it as the evident adjective, or the past tense of the word "bore".

However, because the number of features decline after stemming, prediction accuracy is ought to be gained and computational expense is reduced. These benefits are considered to outweigh the potential information loss suffered by stemming.

[20]The full list of words is found at http://www.lextek.com/manuals/onix/stopwords2.html.

[21]The process of marking words of a sentence with their correct part-of speech-label (i.e. noun, verb, adjective, etc.).

3.6 Data Quality

This thesis is based on secondary data in that data are collected from two external sources to create our data set.

Stock prices are provided from *Center for Research in Security Prices* (CRSP), which is a professional provider of security prices and returns. According to CRSP (n.d.) 500 leading academic institutions in 35 countries rely on CRSP data for academic research, supporting that this is a trustworthy and reliable source of data.

The earnings call transcripts are retrieved from *Seeking Alpha*. They cover 4500 different companies, aiming to provide transcripts six hours after the call is finished. They also have a high focus on error management, aiming to keep errors within a transcript at 0,5% (Alpha, n.d.).

3.6.1 Validity

Validity is a term used to assess how compelling the data set is, and how relevant it is to answer the research question of a thesis (Saunders, Lewis, and Thornhill, 2015). High validity secure that a paper is actually answering the research question.

Validity are usually divided into internal validity and external validity. Internal validity is used to see if causal conclusions can be drawn from a study (Saunders et al., 2015). This thesis is not trying to prove causality between independent and dependent variables, making the demand of high internal validity non existent. External validity concerns how the results of a study can be generalized (Saunders et al., 2015).

The data set contains 3689 different firms, covering companies operating in different industries, subject to different demographics, company cultures and firm sizes. With

this in mind, it should not be problematic to generalize the results of the models to also fit for companies on NYSE and NASDAQ not included in our models.

Generalizing the results to companies that are not listed on NASDAQ and NYSE might be more problematic, due to potential sizable difference in company variables. Another factor that might have an influence is the language barrier. Obviously, the models built in this thesis would struggle when predicting on earnings calls in a different language, but similar results would be expected if this thesis was done with e.g. French transcripts.

3.6.2 Reliability

Reliability is a term used to assess the trustworthiness of a data set. If the results can be replicated by other researchers easily, the results has high reliability (Saunders et al., 2015). Section 4.1 presents the results from a descriptive analysis. The analysis contains a set of figures. Some of these figures are used to evaluate the reliability of the data set.

Figure 10 illustrates that the data set is balanced, meaning that the amount of transcripts labeled up and down is roughly the same. In section 2.5.2 it was suggested that using accuracy as a performance metric is problematic if the data set is unbalanced. Figure 10 supports a claim stating that this is not a problem in this case.

Figure 14 provides insight on how balanced the data set is considering the return of the stock the day after the earnings call. A large portion of the observations is between -1% and 1%, which implies that the classification problem is difficult. A data set with more extreme returns may make it easier for a machine learning algorithm to classify correctly. This is not the case here as the majority of observations fluctuate around 0, making it harder to differentiate the transcripts.

4 Results

This sections presents the results achieved in this thesis. The results are divided in two subsections. The first section is a descriptive analysis of the data set to attain a better understanding of the data and model performance. Following this is the empirical analysis, which displays the performance of the classifiers and whether these are under -or overfitting.

4.1 Descriptive Analysis

This section aims to reveal the reality of the data set. This is done through the analysis of informative figures, tables and interesting findings.

Transcripts by Year

Figure 10 is a bar chart which illustrates how many transcripts are labeled up and down, grouped by year. Color-fillings indicates an upward or downward movement in the stock price following an earnings call conference.

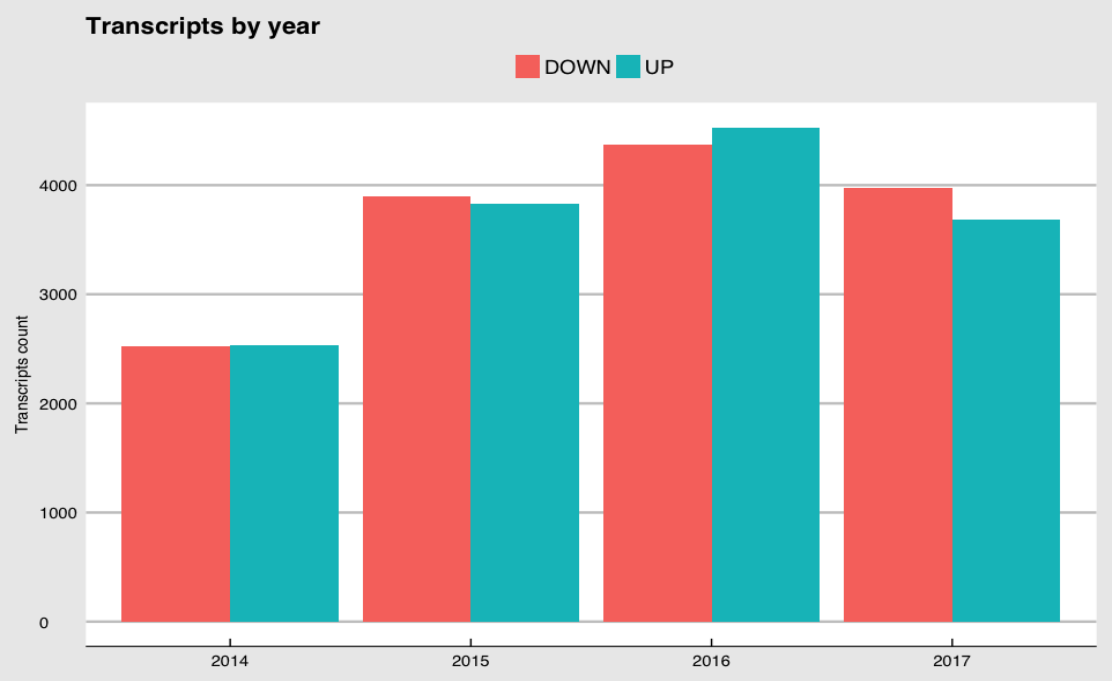


Figure 10: Earnings call transcripts by year and proceeding day's stock price direction

Figure 10 implies that this is a balanced data set. Upwards and downwards movement follows each other relatively closely over the 4 different years. The majority of transcripts are grouped in 2016 and least is grouped in 2014. Across 2015-2017, the transcripts are relatively evenly distributed, with 2014 breaking this pattern.

Table 4: Total stock movement

Movement	Frequency
UP	14,575
DOWN	14,764

Frequent Words

Figure 11 reveals the 40 most frequent terms from the transcripts. This is after the transcripts have been stemmed, cleaned for stopwords and words without information.

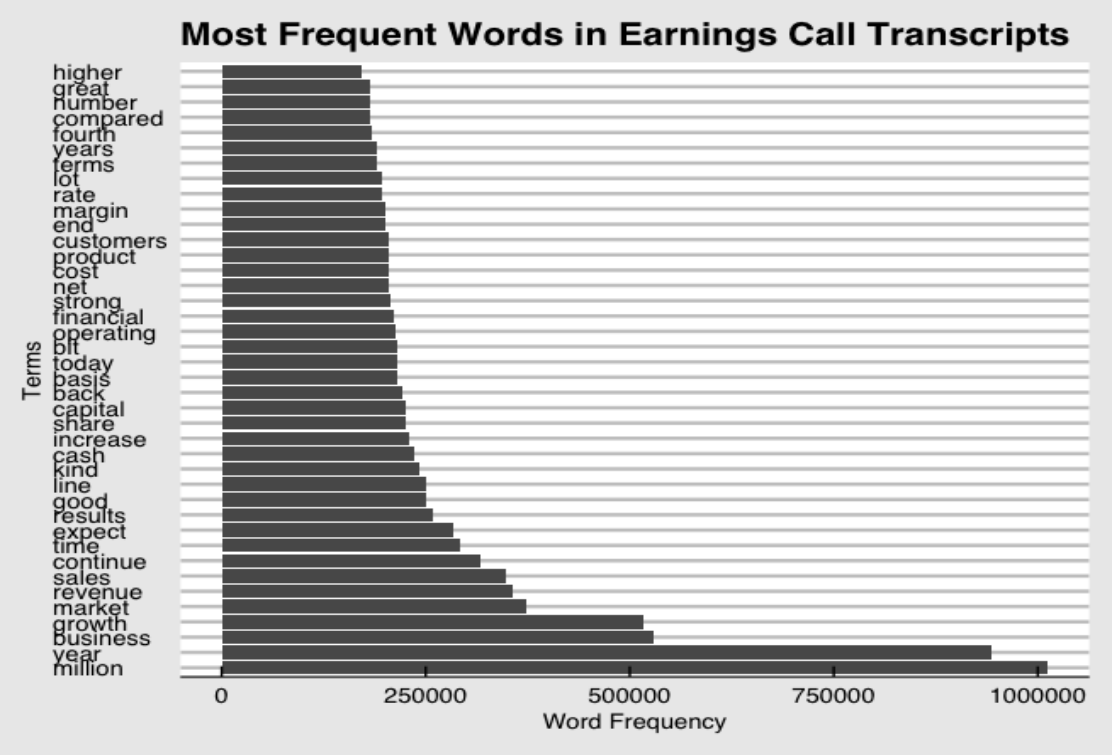


Figure 11: 40 most frequent words in the transcripts

Figure 11 demonstrates how zipf's law presents itself in the transcripts, with the words *million, year, business* and *growth* as the most frequent ones. An interesting observation is that the 3 most frequent terms are associated with downwards movement in stock price. This can be seen by looking for *million, year, business* in figure 12.

Comparison Clouds

Figure 12 is a comparison cloud. It is used to visualize the relationship between words and stock price movement. Words in green are the most frequent words in transcripts that are followed by an upwards stock price movement, while downwards stock price movement is associated with words in orange. The more frequent a word are, the more clearly it is visualized in the word cloud.

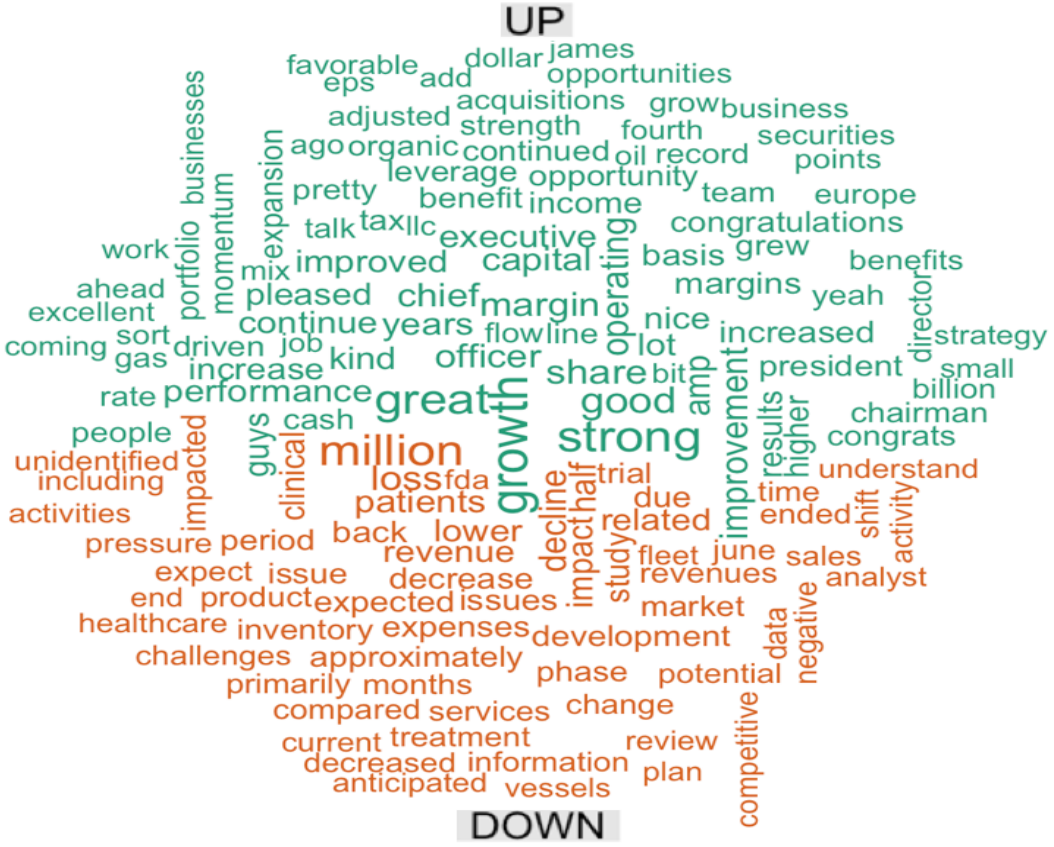


Figure 12: Word cloud based on full transcripts

Figure 12 displays that words as *strong, great, good,* and *improved* are grouped as up, while *loss, difficult* and *issue* are grouped as down. This reveals that there is inherently

different language used in the calls depending on which way the stock is moving subsequently. There are also some odd observations in this word cloud, i.e. *million, revenue* and *product* is grouped as down.

Figure 13 is a second comparison cloud, based only on the Q&A part of the earnings call transcripts.

Figure 13: Word cloud based on Q&A sessions

Compared to figure 12, figure 13 has a lot more prominent words, suggesting that more frequent words might have a stronger impact in a model if the Q&A parts are isolated. An interesting observation is that the word *business* here is a part of the up classification,

in contrast to figure 12. This suggests that the nuances when classifying words could be small and ambiguous.

Levels of Return

Figure 14 is a bar chart, grouping the transcripts by year and the level of the return on the stock, following the earnings conference call.

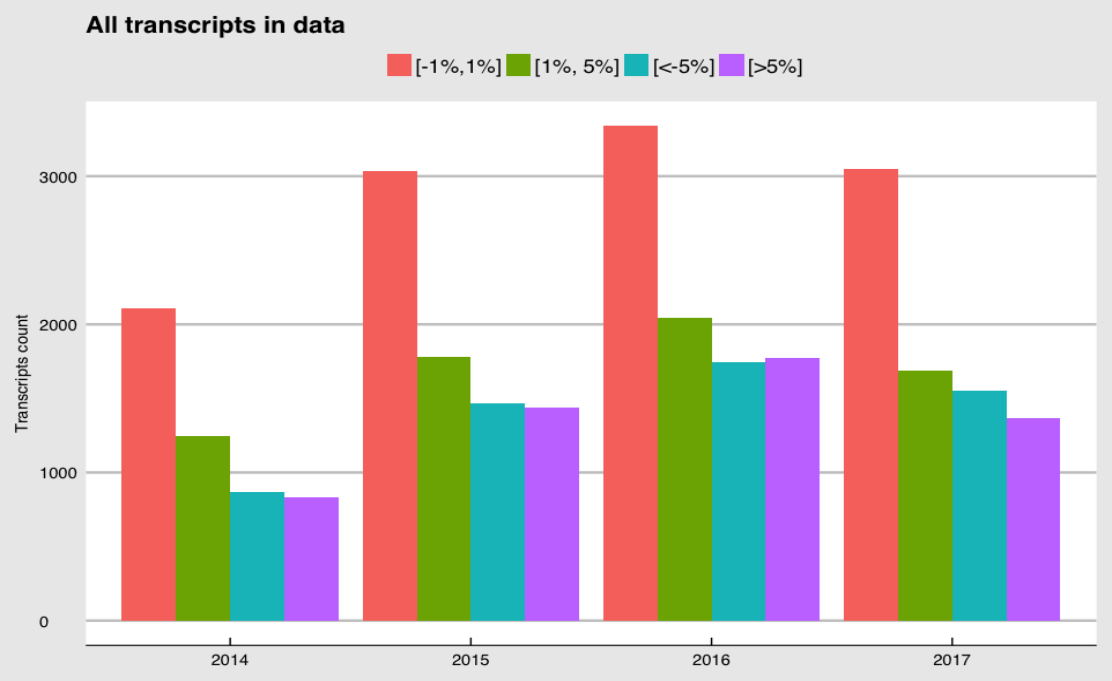


Figure 14: Earnings call transcripts grouped by level of returns

Figure 14 reveals that the majority of transcripts have ”small” changes in return following the earnings call. Small in this case is either from -1% to 1%. This demonstrates the difficulty on the classification problem, with most observations fluctuating around 0%.

Lasso Coefficients

Exploiting the lasso regularization's ability to sort out the most relevant features, table 5 displays the most impactful words in the data set. Words as *impacted, decline, slower* and *disappointing* predict negative impact on the stock price direction following an earnings call. This means that an investor reading or hearing these words during an earnings call should associate this with a higher probability of a negative outcome on the stock movement.

On the other side, *strong, great, congrats* and *improvement* are positive words that predict upward movement. The table demonstrates that lasso regularization does a good job separating words associated with each of the response variable's two labels.

Equation 3.1 illustrates how these words are put to use. Every word from table 5 has a coefficient (x) that makes the expression = *1*. If the classifier sees a word from the table in a transcript, it becomes certain that this transcripts belongs to class up or down, given that it only meets one of these words.

$$Inverse\ Logit\ Function = \frac{e^x}{1+e^x} \tag{4.1}$$

Table 5: Lasso coefficients and impact

Words	Lasso Coefficients	Impact
strong	201.291	Positive
great	163.165	Positive
congrats	143.487	Positive
congratulations	127.043	Positive
improvement	88.923	Positive
improved	79.361	Positive
share	74.525	Positive
nice	68.143	Positive
benefit	33.951	Positive
positively	33.779	Positive
outperformance	28.201	Positive
benefiting	23.064	Positive
sustainable	21.122	Positive
exceptional	18.185	Positive
traditional	17.395	Positive
softness	-23.450	Negative
challenges	-25.829	Negative
disappointed	-28.326	Negative
understand	-28.740	Negative
shift	-31.397	Negative
loss	-36.377	Negative
issue	-37.646	Negative
weakness	-39.806	Negative
delayed	-41.199	Negative
information	-44.579	Negative
changing	-47.934	Negative
disappointing	-51.777	Negative
slower	-53.306	Negative
decline	-53.855	Negative
impacted	-55.855	Negative

Transcript Sentiment

As elucidated in section 2.2.2, sentiment analysis derive the aggregated linguistic tone and mood of a document. The predefined dictionary of Loughran and McDonald (2011) classifies terms as negative, neutral or positive. Running this kind of sentiment analysis on the corpus, table 6 displays the fraction classified as positive versus negative, alongside the price movements the following day for the respective stocks.

An interesting observation is that even though the LM dictionary are accounting for words that otherwise would be considered negative e.g. ”debt”, negative sentiment transcripts outweighs the positive. It is also observed a small positive correlation between sentiment and stock direction, i.e. a transcript with positive (negative) sentiment are weakly associated with an upwards (downwards) stock price movement the subsequent day.

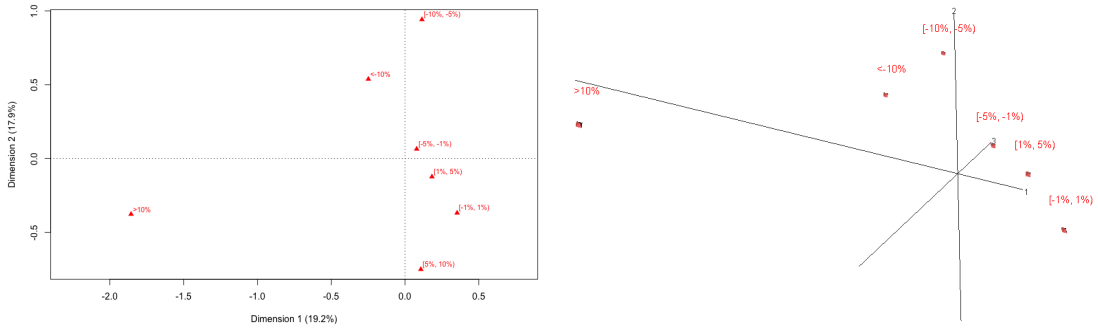
	Sentiment	Stock Direction	Correlation
Positive	12,499	14,750	
Negative	17,178	14,927	0.075

Table 6: Sentiment and next day's stock price movement

Correspondence Analysis and Cosine similarity

In order to assess the similarity between the transcripts, a correspondence analysis is provided. The correspondence analysis is done by ordering the transcripts in seven different groups defined by intervals of return (columns) and comparing this to how many times the different words in the total corpus presents itself in the different groups (rows). By doing this, figure 15 is derived and indicates how the different groups relate to each

other.



(a) Two-dimensional

(b) Three-dimensional

Figure 15: Correspondence analysis

Figure 15a is a correspondence analysis with two dimensions. Dimension one has an explained variance of 19.2% and dimension 2 an explained variance of 17.9%. In total, figure 15a amounts to an explained variance of 37.1%. Groups [1%, 5%), [-5%, -1%) and [-1%, 1%) seems to cluster together which indicates a stronger association between those transcripts. Overall, most groups are to the right of the dimension 2 axis, indicating a larger similarity between these groups.

To investigate the potential relations further, 15b is provided. Figure 15b is a multiple correspondence analysis with three dimensions. This returns a figure with higher explained variance. The figure seems to support the clustering tendencies of groups [1%, 5%), [-5%, -1%) and [-1%, 1%).

In order to further evaluate the similarity between the transcripts, the cosine similarity of the 6 most similar groups relative to the most differing according to the correspondence analysis is presented in table 7. The cosine similarities is displayed in an ascending order from least similar to the most similar group relative to the $>10\%$ group. The

	[-10%,-5%)	[5%,10%)	[-1%,1%)	<-10%	[1%,5%)	[-5%,-1%)	>10%
Cosine sim.	0.170	0.205	0.207	0.236	0.236	0.256	1

Table 7: Cosine similarities relative to the above 10% return group.

cosine similarities illustrate how different the >10% group is to the others. An interesting observation is that the other groups have similar cosine similarities. The groups of [1%,5%) and <-10% are identical, while [-1%,1%) and [5%,10%) are close. Altogether, it seems to be no logical pattern between similarities of transcripts and level of return.

It was anticipated that there would be similarities between the transcripts, however in a more logical manner in which the groups would be ascending in value and sign. This is not the case. In addition, this analysis show that most of the transcripts are very similar, which further suggests that it is challenging for a classifier to find distinct patterns in the data and differentiate our response variable.

4.2 Empirical Analysis

In this section the results from the classifier models will be presented with main emphasis on showing and explaining learning curves for the different algorithms and the respective models. Learning curves is used to display predictive generalization performance and are used to visualize whether or not the models are under -or overfitted to the training data. Classification error plots as a function of number of observations used to train and test the model.

Generally, if the models are of high complexity, it is expected to see an initial decrease of the classification error as the training data increases. Subsequently it is expected to see a point where increasing performance is not possible. If decrease in performance is observed after this point, or if training error is much lower than test error, there is indication of overfitting.[22] The ideal situation would be to see the lowest classification error for the smallest training set. This is firstly because obtaining earning call transcripts for training can be troublesome and expensive, and secondly because of steep increase in computational cost with increasing data.

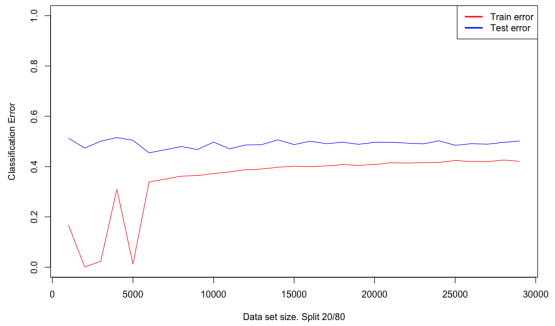
In all of the proceeding plots, the R-package caret[23] is used to train the models. Stochastic Gradient Boosting and Support Vector Machine has more tuning parameter options than naive Bayes and logistic regression. Tuning parameters for these are automatically chosen via caret's operations to find the optimized performance during continuous cross-validation.

All sets are five fold cross-validated and every iteration is accumulated increments of 1000. The training/test split is sampled with replacement 80/20 for every iteration. Red lines plots training error, and blue lines plots test error.

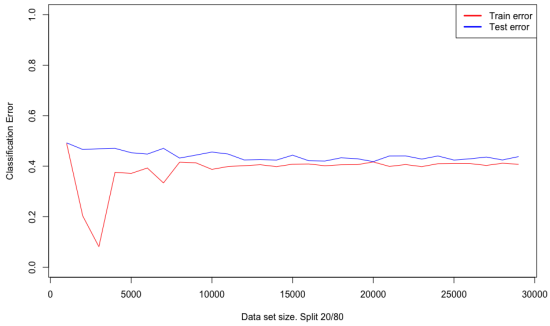
[22] See section 2.4.1.

[23] https://cran.r-project.org/web/packages/caret/index.html

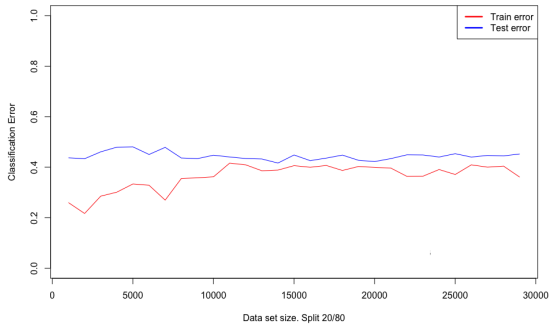
The initial plots show learning curves for the textual classifiers. Subsequent plots show learning curves for benchmark models in which the only feature is the S&P500 index at time *t*, and the response is stock direction for stock *i* at time *t+1*. Lastly an overview of the *corrected resampled t-test* is provided, whereby text classifiers and benchmarks is compared.



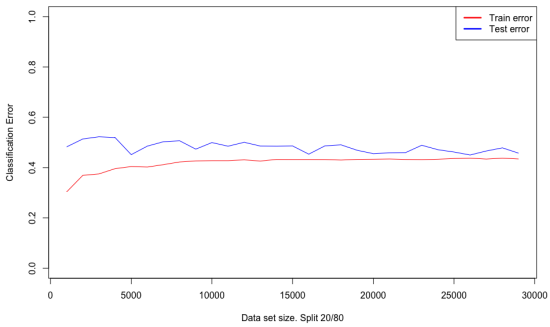
(a) Naive Bayes



(b) Logistic Regression



(c) Stochastic Gradient Boosting



(d) SVM Linear Kernel

Figure 16: Learning curves, textual classifiers

	Naive Bayes	Logistic Regression	Gradient Boosting	SVM
Minimum	0.455	0.418	0.416	0.450
Average	0.491	0.441	0.444	0.481
Full data set	0.501	0.438	0.452	0.457

Table 8: Classification errors

Naive Bayes

Figure 16a display the learning curve for the five fold cross-validated naive Bayes classifier. Average test error is 49.1%, and minimum test error through all 29 iterations is 45.5% when training on 4 800 observations. Notice that there is no sizable performance increase when expanding the data set, and that the curve remains stable at close to 50%. Comparing the training and test error, it is revealed that they both stabilize after the few first iterations. However, there are a sizable gap between them which indicates a high variance classifier and that it overfits the training data.

It is quite evident that there is not much utility in using this particular model for deriving predictive information on market movements from earnings call transcripts. The cause of naive Bayes' poor performance is attributed to the algorithm's simplicity in terms of the naive Bayes assumptions.[24] Contributing to a negative performance is that since naive Bayes is sensitive to correlated features, the fact that words are stemmed leads to no differentiation of inflections and conjugations, the frequency of correlated features then becomes higher which may drive poor performance.

[24] See section 2.3.1

Logistic Regression

Figure 16b plots the learning curve for the logistic regression with lasso regularization. Average test error is 44.1% and minimum is 41.8% when training on 16 000 observations. Training on roughly 23 000 earning calls transcripts yields a test error of 43.8%. It is uncovered that the classification error is stabilizing after the initial iterations, but on a lower level than the other algorithms. A small performance increase is detected when increasing training set size, however this is of no meaningful magnitude when opposing this to how much additional data is handled in each successive iteration.

Explanations for that the logistic regression with lasso regularization is doing better than the other algorithms is simply it's inherent ability to sort out the features that does not have large predictive power. The suggestion from section 2.3.2 seems to be sensible, i.e. that lasso regularization is an easy way of eliminating much of the extreme amounts of noise in textual data simply by forcing the irrelevant words' impact to zero.

Stochastic Gradient Boosting

In figure 16c the learning curve profile is using gradient boosting. Average test error is 44%. Error when training on 80% of the full data set is 45.2%. Looking at how the curve unfolds when expanding the training data, the test-error is stable at around 40-45% through every iteration. We can also observe that in the initial iterations, the training error increases with a larger amount of data.

Allegedly, the stochastic gradient boosting algorithm does fairly well. The caret package chooses tuning parameters that yield a profile without no clear signs of overfitting when expanding the data. This thesis are not able to increase the prediction accuracy beyond the initial 14 iterations whereupon it is observed an accuracy of 58.4%.

Support Vector Machine

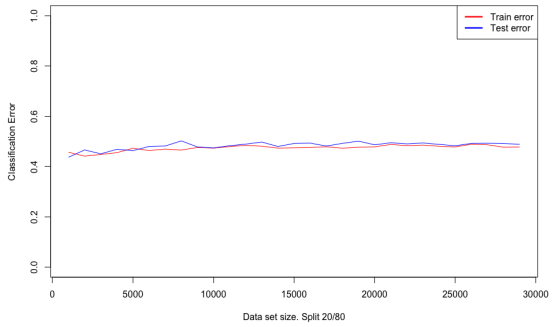
Figure 16d shows the learning curve of the support vector machine with linear kernel. Average test error is 48.1% and for the full training set, test error settle at 45.7%. Looking at the profile of the test error plot, it is detected a somewhat higher variance and a more unclear tendency of the prediction error to stabilize compared to the other algorithms. However, increasing the data set leads to slight increase in performance, whereby minimum training error is 45% training on roughly 21 000 transcripts. It is hard to say whether or not this is due to a gain of increasing training samples or just the outcome of high variance in prediction error. Nevertheless, the increase is small and can hardly justify the incremental data which is needed to produce such a small performance gain.

Benchmarks and Comparison

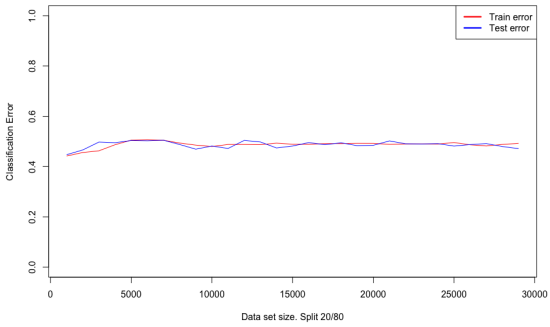
	Naive Bayes	Logistic Regression	Gradient Boosting	SVM
Minimum	0.437	0.447	0.440	0.447
Average	0.483	0.487	0.467	0.486
Full data set	0.489	0.471	0.469	0.482

Table 9: Classification errors benchmark

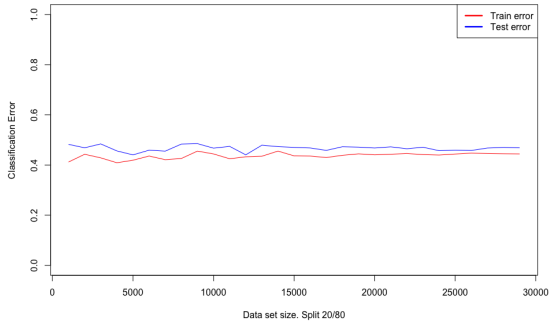
Comparing the learning curves of the benchmarks as opposed to the text classifiers, it is noticed that throughout the benchmark iterations, the training- and test error is more stable and has less deviation between them compared to those of the text classifiers. In addition, poor performance in terms of classification accuracy confirm that the benchmark models undoubtedly are underfitted. This makes sense as the models are only using S&P500 as a predictor, which leads to high bias and thus, underfitting is inevitable.



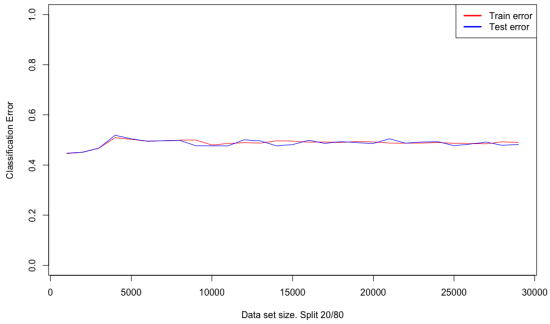
(a) Naive Bayes Benchmark



(b) Logistic Regression Benchmark



(c) Stochastic Gradient Boosting Benchmark



(d) SVM Linear Kernel Benchmark

Figure 17: Learning curves benchmarks

Underfitted models give the impression that they generalize well on unseen data and therefore give way for adjacent and stable learning curves. However they are too simple with regards to the data they are trying to model and in this case. Increase performance by adding complexity would solve this. As the aim of the benchmarks is solely to serve as a means of comparison, adding model complexity is not considered for this paper.

For the text classifiers, the large pool of unique terms included as the training data is allowed to expand, gives rise to high variance and possible overfitting. As mentioned, large gaps between training- and test errors are indication of overfitting. Some indication of this is observed for naive Bayes, but no clear signs for the other algorithms.

Comparing table 8 with table 9, reveals that the naive Bayes model is the only classifier that do worse than the benchmark. For the other models, the mean accuracy is lower, which indicates that they are superior to the benchmark.

To test whether or not the text classifiers are beating the benchmark, a corrected resampled t-test is constructed.[25]. Using the student's t-distribution, we find the upper tail p-values as reported in Table 10. The table show the different components of the t-test, the t-statistics and the respective p-values. From the table, it is inferred that models using logistic regression with lasso regularization, and the gradient boosting are significantly better performing than the benchmark models.

	$\overline{A}$	$\sqrt{\sum_{i=1}^{m} \frac{(A^{(i)}-\overline{A})^2}{m-1}}$	t-statistic	p-value
Naive Bayes	-0.007	0.022	-0.615	0.728
Logistic Regression	0.046	0.025	3.389	0.001
Gradient Boosting	0.023	0.023	1.895	0.034
Support Vector Machine	0.005	0.270	0.385	0.351

Table 10: Corrected resampled t-test, text classification vs S&P500 classifier. Df=28 (see section 3.3.2 for definitions)

[25] Outlined in section 3.3.2

5 Discussion

This thesis has looked into the predicting power of earnings call transcripts on stock price movement the following day. The results demonstrates that logistic regression, gradient boosting and SVM models based on transcripts beat their corresponding benchmarks based on S&P500. Of these three, logistic regression and gradient boosting is significantly different from their corresponding benchmark models. Naive Bayes fails to beat its benchmark and with a test error of 49,1% struggles to recognize a pattern in the data set. However, there are many factors that influence the results of this thesis and this section will discuss factors that might explain the results.

Although the results are trustworthy, it is hard to rate the results achieved in this thesis. There is a small amount of similar projects to compare with, and the ones that can be found are questionable with either very good or very bad results. In his master thesis, Liang (2016) uses earnings call transcripts of major U.S airlines and all his models fail to result accuracy over 50%. On the other hand, Ulrich, Pratt, and Thun-Hohenstein (2016) consistently achieves accuracy between 70-75% using Naive Bayes to predict stock price movement using earnings call transcripts. They have a substantial longer response time on their response variable. Considering this, it appears as the results in this thesis is decent, and it seems like it is possible to notably improve the results.

An argument that needs to be considered when justifying the results are the benchmarks. Beating the benchmark means that the text classifiers is better in predicting stock price movement than predictive models trained on previous days' S&P500 index levels. The intuition behind this benchmark can be found in section 3.3.2. In this case, three of four models outperform their benchmark. However, this might not be evident proof of powerful models. Figure 17 shows that the benchmark learning curves from the test set and training set is following each other very closely, which indicates underfitted models. A

reasonable presumption is that these benchmark models are in any manner easy to beat. Benchmarks with more features and reasonable fit would possibly result in a more fair comparison against the models fitted to the transcripts. This would lead to more impactful results in the cases where the benchmark is beaten.

Generally, it is expected that machine learning models will improve performance as more data is provided. None of the learning curve plots in chapter 4 demonstrates clear signs of better performance when more data is included. This indicates that the classifiers are not complex enough to learn explicit clear cut patterns between the words used in an earnings call transcript and the direction of stock prices the subsequent day. It would be interesting to see what more complex algorithms could achieve on this data set. However, it is reasonable to believe that more complex algorithms also would struggle to pick up signals from the data set, as is discussed in section 3.1.

With all the words and features in the data set, it seems likely that it is hard for the algorithms to separate the signals from noise in the data set, given that there actually are some pattern to pick up. One explanation for this might be due to bag of words' approach generating far to many features, making it hard to detect possible incremental value in the transcripts. According to Mikolov and Le (2014) the bag of words approach have two extensive weaknesses, there are no ordering of the words and the semantics of the words are ignored. These are important aspects, and it is possible that trading the semantics of words and sentences for the simplicity of bag of words is an unreasonable cost when using transcripts for predictions. More impressive results might be achieved either by utilizing a better way for feature selection with bag of words customized for earnings call transcripts, or through implementing word embeddings as the word2vec model of Mikolov, Sutskever, Chen, Corrado, and Dean (2013).

In general, the lack of understanding semantics and complete sentences leads to further

problems. The data set is bound to have companies that are negatively correlated with respect to stock price, and the amount of negatively correlated stocks could make it hard to detect a pattern with words as features. To exemplify the problem consider the word decline. A transcript discussing decline in oil prices would obviously be negative in a oil company, while for an airline this would be considered good news. Without a feature that considers the companies and their corresponding industries, it is hard to catch this effect in a model and in general this substantiates the information loss when treating every single word as a feature. A possible solution to this could be to refine the classification task to only one industry.

Other aspects that might be of influence on performance are companies' differences of well known risk factors i.e. the Fama–French three-factor model (Fama and French, 1993). With this in mind, the influence of words might end in different results. For example, it is reasonable to expect that the market would weigh linguistic report of the previous quarter differently between companies with a large asset base, and companies where the majority of the cash flow are expected in the future. Negative wording about a short time interval coming from growth companies might have less impact on short term stock movements if the market believes in future growth. An interesting refinement in regards to the research question would thus be to see whether performance would increase predicting only on companies with similar accounting metrics, such as book to market ratios.

A potential view on this thesis' results is that the best results possible is achieved, quite simply because there is no big predictive pattern to detect. Earnings calls is of a very repetitive format and it is possible that a big proportion of earnings call transcripts are to similar, something the correspondence analysis and cosine similarity suggest. Although each company has its own agenda, they mostly follow the same formula, using the same repeated lines and buzzwords, which creates a big proportion of noise. This

could be a potential explanation for why it is hard to make a high performing classifier on earning calls transcripts. This would be in line with what Matsumoto et al. (2011) demonstrate, the Q&A session of an earnings call is more informative. This is probably due to the more spontaneous responds, generating differences from the usual well prepared presentation part. Isolating the Q&A part of an earnings call transcript to use for predictions might return better results.

Furthermore it could be questioned whether there is significant value and predictiveness in the Q&A part. Mostly, the questions are asked from brokerage firms, which should be treated with healthy skepticism from investors. Brokerage firms contributes with sell side analysts, which might use their questions to enhance their recommendation models with a shorter view than the general investor. Also, Cohen, Lou, and Malloy (2014) finds in their paper that company management have sell side analysts they favour and like, which are invited to ask questions and manipulate the earnings call. In addition, buy-side analysts, arguably the most important participants of the call, are most likely reluctant to reveal their thinking, and for this reason will primarily just be listening. If this is the case, most investors would not act as strongly upon the information in the Q&A session, and in this way hamper distinct stock price movement that can be picked up by the models.

6 Conclusion

The purpose of this thesis is to investigate the predicting power of earnings call transcripts on stock price movement. The introduction provides a literature review concerning earnings calls and suggests that earning call transcripts contains valuable incremental information, which can be extracted in different ways. This is in line with the results of this thesis.

The thesis infers that there are predicting power in earnings call transcripts, given use of an appropriate classifier. Of total four different classifiers, two are significantly better performing than their corresponding benchmark. The best performing model is logistic regression, which achieves an average accuracy of 55.9%. Following this is gradient boosting with an accuracy of 55.6%. The SVM models beats its benchmark and attains an accuracy of 51.9%, however this result fails to be significantly different from the corresponding benchmarks performance. Lastly, naive Bayes fails to beat its benchmark and obtain an average accuracy of only 50.9%.

The thesis makes a helpful contribution in demonstrating how textual analysis and machine learning can be used in finance. Textual analysis in finance is an emerging discipline and this thesis show how relatively inexperienced students can develop decent performing predicting classifiers. This suggest that there might be possible to pick low hanging fruits from doing further experiments and research, in that developing something useful is not alarmingly unrealistic or hard. In the end, this could lead to reduced investor costs or deeper understanding of the financial field.

The results expose that there are predictive power in earnings call transcripts to some degree. These results can probably be improved, given technological advances or potential smarter solutions than used in this thesis. The thesis does not have a satisfactory

answer to the amount of predictive power in an earnings call transcript. To this, the benchmarks used to evaluate performance are considered too simple, which makes the results less impactful.

A potential weakness of this thesis is the utilization of the bag of words approach, which seems to be a double edged sword. Its simplicity makes answering the research question more feasible, still the drawbacks might be outweighing the perks. The simplicity appears to hamper the performance of classifiers by disregarding that words and phrases might have several meanings. In addition, the descriptive analysis with figure 14, figure 15 and table 7, demonstrates the difficulty of the classification problem, suggesting that developing a high performing classifier might be to ambitious.

Our work with this particular research question has made us notice other ways that might elucidate the predictive power of earnings call transcripts. Firstly, we would suggest to illuminate the classification problem by delineate the selection of companies by choosing companies with similar book to market ratios or companies from only one industry. Secondly we would suggest predicting using only the Q&A session of an earnings call. Thirdly, if possible, we would suggest using more models. Lastly, we would suggest to investigate the possibilities with word2vec and similar word embedding models.

Finally we would urge further researches to focus on falsifying economic and financial theories. After working with this research question we agree and understand Li (2010)'s arguments that too much of the literature are focusing on finding ways to apply off-the-shelf textual methods borrowed from highly evolved technologies in computational linguistics. It would be more beneficial with more research motivated by hypotheses closely tied to economic theories.

References

Aalst, W. M. P. van der, V. Rubin, H. M. W. Verbeek, B.F. van Dongen, E. Kindler, and C.W. Günther (2010). "Process Mining: a Two-Step Approach to Balance Between Underfitting and Overfitting". *Software & Systems Modeling* 9, pp. 87–111. DOI: 10.1007/s10270-008-0106-z.

Alpha, Seeking (n.d.). *SA Transcripts*. URL: https://seekingalpha.com/page/sa_transcripts. (accessed: 23.04.2018).

Antweiler, Werner and Murray Z. Frank (2004). "Is All That Talk Just Noise? The Information Content of Internet Stock Message Boards". *Journal of Finance* 59.3, pp. 1259–1294. DOI: 10.1111/j.1540-6261.2004.00662.x.

Arlot, Sylvain and Alain Celisse (2010). "A Survey of Cross-Validation Procedures for Model Selection". *Statistics Surveys* 4, pp. 40–79. DOI: 10.1214/09-SS054.

Baker, Malcolm and Jeffrey Wurgler (2007). "Investor Sentiment in the Stock Market". *Journal of Economic Perspectives* 21.2, pp. 129–152. DOI: 10.1257/jep.21.2.129.

Bloomfield, Robert (2008). "Discussion of "Annual Report Readability, Curren Earnings, and Earnings Persistence". *Journal of Accounting and Economics* 45.2-3, pp. 248–252. DOI: 10.1016/j.jacceco.2008.04.002.

Borra, Simone and Agostino Di Ciaccio (2010). "Measuring the Predcition Error. A Comparison of Cross-Validation, Bootstrap and Covariance Penalty Methods". *Computaational Statistics & Datas Analysis* 54.12, pp. 2976–2989. DOI: 10.1016/j.csda.2010.03.004.

Bowen, Robert M., Angela K. Davis, and Dawn A. Matsumoto (2001). "Do Conference Calls Affect Analysts' Forecasts". *The Accounting Review* 77.2, pp. 285–316. DOI: 10.2139/ssrn.216810.

Buehlmaier, M. and T. M. Whited (2014). "Looking for Risk in Words: A Narrative Approach to Measuring the Pricing Implications of Finance Constraints". Working paper, University of Rochester, 2014. URL: http://hdl.handle.net/10722/213634.

Buehlmaier, M. and J. Zechner (2013). "Slow-Moving Real Information in Merger Arbitrage". Working paper, University of Hong Kong, 2013. URL: http://hdl.handle.net/10722/201724.

Clausen, Stein Erik (1998). *Applied Correspondence Analysis*. SAGE Publications, Inc. ISBN: 9780761911159.

Cohen, Lauren, Dong Lou, and Christopher Malloy (2014). "Playing Favorites: How Firms Prevent the Revelation of Bad News". Working Paper, Harvard Business School. URL: http://www.people.hbs.edu/lcohen/pdffiles/coh_lou_mal.pdf.

Core, John E. (2001). "A Review of the Empirical Disclosure Literature: Discussion". *Journal of Accounting and Economics* 31.1-3, pp. 441–456. DOI: 10.1016/S0165-4101(01)00036-2.

Cortes, Corinna and Vladimir Vapnik (1995). "Support-Vector Networks". *Machine Learning* 20.3, pp. 273–297. DOI: 10.1007/BF00994018.

CRSP (n.d.). *About CRSP*. URL: http://www.crsp.com/about-crsp. (accessed: 23.04.2018).

Das, Sanjiv Ranjan (2014). "Text and Context: Language Analytics in Finance". *Foundations and Trends in Finance* 8.3, pp. 145–261. DOI: 10.1561/0500000045.

Davis, Angela K., Jenny L. Zhang, Weili. Ge, and Dawn Matsumoto (2015). "The Effect of Manager-Specific Optimism on the Tone of Earnings Conference Calls". *Review of Accounting Studies* 20.2, pp. 639–673. DOI: 10.1007/s11142-014-9309-4.

Fama, Eugene F. and Kenneth R. French (1993). "Common risk factors in the returns on stocks and Bonds". *Journal of Financial Economics* 33.1, pp. 3–56. DOI: 10.1016/0304-405X(93)90023-5.

Friedman, Jerome H. (2002). "Stochastic Gradient Boosting". *Computational Statistics & Data Analysis* 38.4, pp. 367–378. DOI: 10.1016/S0167-9473(01)00065-2.

Friedman, Jerome H., Trevor Hastie, and Robert Tibshirani (2000). "Additive Logistic Regression: a Statistical View of Boosting". *The Annals of Statistics* 28.2, pp. 337–407. DOI: 10.1.1.51.9525.

Gaikwad, Sonali V., Archana Chaugule, and Pramod Patil (2014). "Text Mining Methods and Techniques". *International Journal of Computer Applications* 85.17, pp. 42–45. DOI: 10.5120/14937-3507.

German, Stuart, Elie Bienenstock, and René Doursat (1992). "Neural Networks and the Bias/Variance Dilemma". *Neural Computation* 4.1, pp. 1–58. DOI: 10.1162/neco.1992.4.1.1.

Greenacre, Michael and Jörg Blasius (1994). *Correspondence Analysis in the Social Sciences*. Academic Press. ISBN: 9780121045708.

Greenacre, Michael and Trevor Hastie (1987). "The Geometric Interpretation of Correspondence Analysis". *Journal of the American Statistical Association* 82.398, pp. 437–447. DOI: 10.2307/2289445.

Guay, Wayne., Delphine Samuels, and Daniel Taylor (2016). "Guiding trough the Fog: Financial Statement Complexity and Voluntary Disclosure". *Journal of Accounting and Economics* 62.2-3, pp. 234–269. DOI: 10.1016/j.jacceco.2016.09.001.

Guoa, Li, Feng Shib, and Jun Tua (2016). "Textual Analysis and Machine Learning: Crack Unstructured Data in Finance and Accounting". *The Journal of Finance and Data Science* 2.3, pp. 153–170. DOI: 10.1016/j.jfds.2017.02.001.

Hardeniya, Nitin, Jacob Perkins, Deepti Chopra, Nisheeth Joshi, and Iti Mathur (2016). *Natural Language Processing: Python and NLTK*. Packt. ISBN: 9781787287846.

Hastie, Trevor, Robert Tibshirani, and Jerome Friedman (2009). *The Elemens of Statistical Learning: Data Mining, Inference, and Prediction (Second Edition)*. Springer. ISBN: 9780387848587.

Henry, Elaine (2006). "Market Reaction to Verbal Components of Earnings Press Releases: Event Study Using a Predicative Algorithm". *Journal of Emerging Technologies in Accounting* 3.1, pp. 1–19. DOI: 10.2308/jeta.2006.3.1.1.

Hirschfeld, H.O (1935). "A Connection Between Correlation and Contingency". *Proceedings of the Cambridge Philosophical Society* 31.4, pp. 520–524. DOI: 10.1017/S0305004100013517.

Hoberg, Gerard and Gordon Phillips (2016). "Text-Based Network Industries and Endogenous Product Differentiation". *Journal of Political Economy* 124.5, pp. 1423–1465. DOI: https://doi.org/10.1086/688176.

Hsu, Chih-Wei, Chih-Chung Chang, and Chih-Jen Lin (2016). "A Practical Guide to Support Vector Classification". Guide, National Taiwan University. URL: https://www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf.

Huang, Allen, Amy Zang, and Rong Zheng (2014a). "Evidence on the Information Content of Text in Analyst Reports". *The Accounting Review* 89.6, pp. 2151–2180. DOI: 10.2308/accr-50833.

Huang, Anna (2008). "Similarity Measures for Text Document Clustering". *Proceedings of the 6th New Zealand Computer Science Research Student Conference*.

Huang, Xuan, Siew H. Teoh, and Yinglei Zhang (2014b). "Tone Management". *The Accounting Review* 89.3, pp. 1083–1113. DOI: 10.2308/accr-50684.

IDC (2017). “Data Age 2025: The Evolution of Data to Life Critical”. IDC, Seagate. URL: https://www.seagate.com/files/www-content/our-story/trends/files/Seagate-WP-DataAge2025-March-2017.pdf.

Jones, Karen S. (1972). “A Statistical Iinterpretation of Term Specificity and its Application in Retrieval”. *Journal of Documentation* 28.1, pp. 11–21. DOI: 10.1108/eb026526.

Jones, Michael J and Paul A. Shoemaker (1994). “Accounting narratives: A Review of Empirical Studies of Content and Readability”. *Journal of Accounting Literature* 13.1, pp. 142–184. URL: https://search.proquest.com/docview/216304635?accountid=37265.

Jurafsky, Daniel and James H. Martin (2017). *Speech and Language Processing, third edition*. https://web.stanford.edu/ jurafsky/slp3/. ISBN: 978-0131873216.

Kearney, Colm and Sha Liu (2014). “Textual Sentiment in Finance: A Survey of Methods and Models”. *International Review of Financial Analysis* 33, pp. 171–185. DOI: 10.1016/j.irfa.2014.02.006.

Khan, Aurangzeb, Baharum Baharudin, Lam H. Lee, and Khairullah Khan (2010). “A Review of Machine Learning Algorithms for Text-Documents Classification”. *Journal of Advances in Information Technology* 1.1, pp. 4–20. DOI: 10.4304/jait.1.1.4-20.

Kimbrough, Michael D. (2005). “The Effect of Conference Calls on Analyst and Market Underreaction to Earnings Announcements”. *The Accounting Review* 80.1, pp. 189–219. URL: http://www.jstor.org/stable/4093166.

Kwartler, Ted (2017). *Text mining in practice with R*. John Wiley & Sons, Inc. ISBN: 1119282012.

Larcker, David F. and Anastasia A. Zakolyukina (2012). “Detecting Deceptive Discussions in Conference Calls”. *Journal of Accounting Research* 50.2, pp. 495–540. DOI: 10.1111/j.1475-679X.2012.00450.x.

Lebar, Mary A. (1982). “A General Semantics Analysis of Selected Sections of the 10-K, the Annual Reprot to Shareholders, and the Financial Press Release”. *The Accounting Review* 57.1, pp. 176–189. URL: http://www.jstor.org/stable/246748.

Leuz, Christian and Cathrine Schrand (2009). “Disclosure and the Cost of Capital: Evidencefrom Firms’ Responses to the Enron Shock”. Working Paper, University of Chicago. URL: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=1319646.

Leuz, Christian and Peter D. Wysocki (2016). "The Economics of Disclosure and Financial Reporting Regulation: Evidence and Suggestions for Future Research". *Journal of Accounting Research* 54.2, pp. 525–622. DOI: 10.1111/1475-679X.12115.

Li, Feng (2008). "Annual Report Readability, Current Earnings, and Earnings Persistence". *Journal of Accounting and Economics* 45.2-3, pp. 221–247. DOI: 10.1016/j.jacceco.2008.02.003.

— (2010). "The Information Content of Forward-Looking Statements in Corporate Filings—A Naïve Bayesian Machine Learning Approach". *Journal of Accounting Research* 48.5, pp. 1049–1102. DOI: 10.1111/j.1475-679X.2010.00382.x.

Liang, Dong (2016). "Predicting Stock Price Changes with Earnings Call Transcripts". Master Thesis, University of North Carolina. URL: https://cdr.lib.unc.edu/indexablecontent/uuid:65ad9d24-b9db-4002-8101-38dad962acee. (accessed: 22.05.2018).

Loughran, Tim and Bill McDonald (2011). "When Is a Liability Not a Liability? Textual Analysis, Dictionaries, and 10-Ks". *Journal of Finance* 66.1, pp. 35–65. DOI: 10.1111/j.1540-6261.2010.01625.x.

— (2014). "Measuring Readability in Financial Disclosures". *Journal of Finance* 69.4, pp. 1643–1671. DOI: 10.1111/jofi.12162.

— (2016). "Textual Analysis in Accounting and Finance: A Survey". *Journal of Accounting Research* 54.4, pp. 1187–1230. DOI: 10.1111/1475-679X.12123.

Loughran, Tim, Bill McDonald, and Hayong Yun (2008). "A Wolf in Sheep's Clothiing: The Use of Ethics-Related Terms in 10-K Reports". *Journal of Business Ethics* 89.1, pp. 39–49. DOI: 10.1007/s10551-008-9910-1.

Manning, Christofer D. and Hinrich Schütze (2003). *Foundations of Statistical Natural Language Processing*. Cambridge, MA: MIT Press.

Maron, M. E. (1961). "Automatic Indecing: An Experimental Inquiry". *Journal of ACM (JACM)* 8.3, pp. 404–417. DOI: 10.1145/321075.321084.

Matsumoto, Dawn, Maarten Pronk, and Erik Roelofsen (2011). "What Makes Conference Calls Useful? The Information Content of Managers' Presentations and Analysts Discussion Sessions". *The Accounting Review* 86.4, pp. 1383–1414. DOI: 10.2308/accr-10034.

McCallum, Andrew and Kamal Nigam (1998). "A Comparison of Event Models for Naïve Bayes Text Classification". In AAAI-98 Workshop on learning for text categorization. URL: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.65.9324.

Metz, Charles E. (1978). "Basic Principles of ROC Analysis". *Seminars in Nuclear Medicin* 8.4, pp. 283–298. DOI: 10.1016/S0001-2998(78)80014-2.

Mikolov, Tomas and Quoc Le (2014). "Distributed Representations of Sentences and Documents". *Proceedings of Machine Learning Research* 32.2, pp. 1188–1196.

Mikolov, Tomas, Ilya Sutskever, Kai Chen, G.s Corrado, and Jeffrey Dean (2013). "Distributed Representations of Words and Phrases and their Compositionality". *Advances in Neural Information Processing Systems* 26.

Minsky, Marvin (1961). "Steps toward Artificial Intelligence". *Proceedubgs of the IRE* 49.1, pp. 8–30. DOI: 10.1109/JRPROC.1961.287775.

Mosteller, Frederick and David L. Wallace (1963). "Inference in an AUthorship Problem: A Comparative Study of Discrimination Methods Applied to the Authorship of the Disputed Federalist Papers". *Journal of the American Statistical Association* 302.58, pp. 275–309. DOI: 10.1080/01621459.1963.10500849.

Nadeau, Claude and Yoshua Bengio (2003). "Inference for the Generalization Error". *Machine Learning* 52.3, pp. 239–281. DOI: 10.1023/A:1024068626366.

Natekin, Alexey and Alois Knoll (2013). "Gradient Boosting Machines- a Tutorial". *Front Neurorobot* 21.7. DOI: 10.3389/fnbot.2013.00021.

Pennsylvania, University of (n.d.). *Overfitting and Regularization*. URL: https://alliance.seas.upenn.edu/~cis520/dynamic/2017/wiki/index.php?n=Lectures.Overfitting. (accessed: 09.05.2018).

Price, McKay S., James S. Doran, and David R. Peterson (2010). "Earnings Conference Call Content and Stock Price: The Case of REITs". *Journal of Real Estate Finance and Economics* 45.2, pp. 402–434. DOI: 10.1007/s11146-010-9266-z.

Price, McKay S., James S. Doran, David R. Peterson, and Barbara A. Bliss (2012). "Earnings Conference Calls and Stock Returns: The Incremental Informativeness of Textual Tone". *Journal of Banking and Finance* 36.4, pp. 992–1011. DOI: 10.1016/j.jbankfin.2011.10.013.

Reyes, Antonio and Paolo Rosso (2011). "Mining Subjective Knowledge From Customer Reviews: A Specific Case of Irony Detection". Conference Paper, Association for Computatuinal Linguistics. URL: http://www.aclweb.org/anthology/W11-1715.

Salton, Gerard and Christopher Buckley (1988). "Term-Weighting Approaches in Automatic Text Retrieval". *Information Processing and Management* 24.5, pp. 513–523. DOI: 10.1016/0306-4573(88)90021-0.

Saunders, Mark N.K., Philip Lewis, and Adrian Thornhill (2015). *Research Methods for Business Students*. Pearson Education Limited. ISBN: 9781292016641.

Skianis, Konstantinos, Francois Rousseau, and Michalis Vazirgiannis (2016). "Regularizing Text Categorization with Clusters of Words". *Conference paper: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. DOI: 10.18653/v1/D16-1188.

Smith, James E. and Nora P. Smith (1971). "Readability: A Measure of the Performace of the Communication Function of Financial Reporting". *The Accounting Review* 46.3, pp. 552–561. URL: http://www.jstor.org/stable/pdf/244524.

Stapor, Katarzyna (2017). "Evaluating and Comparing Classifiers: Review, Some Recommendations and Limitations". *Proceedings of the 10th International Conference on Computer Recognition Systems CORES 2017*, pp. 12–21. DOI: 10.1007/978-3-319-59162-9.

Subramanian, Ram, Robert G. Insley, and Rodney D. Blackwell (1993). "Performance and Readability: A Comparison of Annual Reports of Profitable and Unprofitable Corporations". *Journal of Business Communication* 30.1, pp. 49–61. DOI: 10.1177/002194369303000103.

Tasker, Sarah C. (1998). "Bridging the Information Gap: Quarterly Conference Calls as a Medium for Voluntary Disclosure". *Review of Accounting Studies* 3.1-2, pp. 137–167. DOI: 10.1023/A:1009684502135.

Tetlock, Paul. C (2007). "Giving Content to Investor Sentiment: The Role of Media in the Stock Market". *Journal of Finance* 62.3, pp. 1139–1168. DOI: 10.1111/j.1540-6261.2007.01232.x.

Thorsten, Joachims (1998). "Text Categorization with Support Vector Machines: Learning with Many Relevant Features". *ECML'98 Proceedings of the 10th European Conference on Machine Learning*, pp. 137–142. DOI: 10.1007/BFb0026683.

Tibshirani, Robert (1996). "Regression Shrinkage and Selection via the Lasso". *Journal of the Royal Statistical Society* 58.1, pp. 267–288.

Tušar, Tea, Klemen Gantar, Valentin Koblard, Bernard Ženko, and Bogdan Filipiča (2017). "A Study of Overfitting in Optimization of a Manufacturing Quality Control Procedure". *Applied Soft Computing* 59, pp. 77–87. DOI: 10.1016/j.asoc.2017.05.027.

Twedt, Brady and Lynn Rees (2012). "Reading Between the Lines: An Empirical Examiniation of Qualitative Attributes of Financial Analysts' Reports". *Journal of Ac-*

counting and Public Policy 31.1, pp. 1–21. DOI: 10.1016/j.jaccpubpol.2011.10.010.

Ulrich, Thomas, Chaz Pratt, and Phillip Thun-Hohenstein (2016). "Machine Learning Analysis of Company Earnings Releases". CS229, Standford. URL: http://cs229.stanford.edu/proj2016/poster/UlrichPrattThunhohenstein-MachineLearningAnalysisOfCompanyEarningsReleases-poster.pdf. (accessed: 22.05.2018).

Zobel, Justin and Alistair Moffat (1998). "Exploring the similarity space". *Information Processing and Management* 32.1, pp. 18–34. DOI: 10.1145/281250.281256.