BRUNEL UNIVERSITY LONDON

COLLEGE OF ENGINEERING, DESIGN AND PHYSICAL SCIENCES

DEPARTMENT OF COMPUTER SCIENCE

DOCTOR OF PHILOSOPHY DISSERTATION

SENTIMENT ANALYSIS: TEXT PRE-PROCESSING, READER VIEWS AND CROSS DOMAINS

BY

EMMA HADDI

MARCH 2015

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

Reading for a PhD was a long journey full of exciting as well as challenging moments. Many of those would not have been possible come over with. I would like to thank my supervisor Prof. Xiaohui Liu for his support during this time. I also would like to thank my second supervisor Dr. Timothy Cribbin for all the help and useful tips he provided me with.

I owe a great deal to my my family that has given me a great support throughout my PhD studies. To my father Saied Haddi whom his warm eyes and big heart were always in my mind despite the great distance. To my mother Sahar Baradie who is my role model in life with her determination and unconditional giving. To my sister, the family philosopher, Jullanar Haddi who is my best friend and the closest to my heart. Despite the circumstances that kept us drifted apart, they were always there for me. To them I say: I love you and miss you all, and pray for peace in my beloved country Syria so that I can see you again.

I also would like to thank Heidelotte Hoppe for all of her support. With her overwhelming love and kindness, she gave me many warm moments and compensated me for the absence of my parents.

Lastly, I would like to dedicate this work to the love of my life, Konrad Hoppe. Step by step, we were together all the way in our PhD studies. We shared the good and bad moments. He has been and always will be the light in my darkest times, and the hope I hold on to when I need him. I do not know what to say more than that he is the best thing that ever happened to me, and this PhD would not have reached an end without him being by my side all along. I love you Konrad.

PUBLICATIONS

Haddi, E., Liu, X. and Shi, Y. (2013), The role of text pre-processing in sentiment analysis, Procedia Computer Science , Vol. 17, Elsevier, p. 26.

Haddi, E., Cribbin, T., Shi, Y., Liu, X. (2015) Analysing sentiment from financial news to movie reviews: a computational framework, International Journal of Information Technology & Decision Making, (Submitted).

ABSTRACT

Sentiment analysis has emerged as a field that has attracted a significant amount of attention since it has a wide variety of applications that could benefit from its results, such as news analytics, marketing, question answering, knowledge management and so on. This area, however, is still early in its development where urgent improvements are required on many issues, particularly on the performance of sentiment classification. In this thesis, three key challenging issues affecting sentiment classification are outlined and innovative ways of addressing these issues are presented. First, text pre-processing has been found crucial on the sentiment classification performance. Consequently, a combination of several existing pre-processing methods is proposed for the sentiment classification process. Second, text properties of financial news are utilised to build models to predict sentiment. Two different models are proposed, one that uses financial events to predict financial news sentiment, and the other uses a new interesting perspective that considers the opinion reader view, as opposed to the classic approach that examines the opinion holder view. A new method to capture the reader sentiment is suggested. Third, one characteristic of financial news is that it stretches over a number of domains, and it is very challenging to infer sentiment between different domains. Various approaches for cross-domain sentiment analysis have been proposed and critically evaluated.

CHAPTER 1

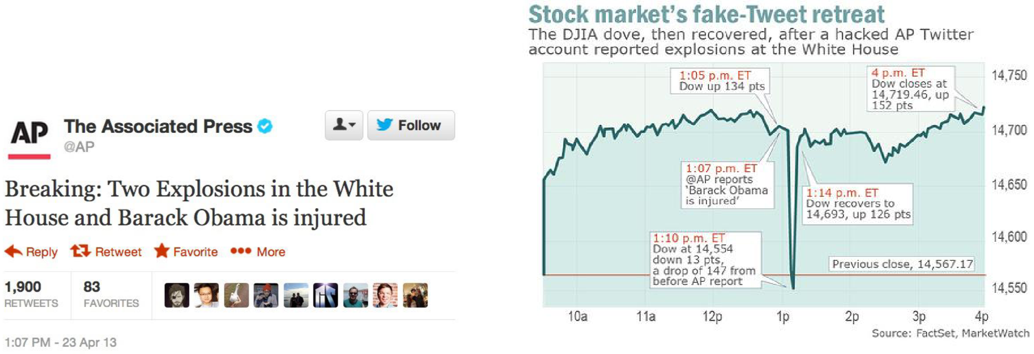
INTRODUCTION

Sentiment analysis is a research field that has received a considerable attention in the last decade. This field comprises a wide range of different applications that have been addressed in many different research studies. In recent years, much effort has been made to investigate the impacts of various means of media on the financial world. The Internet has become a huge source of all sort of information for everyone. Several types of information are of interest in the financial world. For example, investors are concerned about financial news that is related to their investment. Companies are interested in the news about competitors, suppliers, materials, and feedback from customers. Customers in return are interested in other customer reviews about products that they want to purchase. To cater this, various applications of sentiment analysis have emerged in several domains like sentiment analysis of financial news, products reviews, political elections and healthcare [71, 82].

Twitter, one of the most popular social media blogs, is gaining much attention in the recent literature [9, 33, 42, 120, 119]. The investigation of the sentiment of Twitter data or the so called the public mood is used in several fields and applications, such as behavioural economics with applications to the stock market [18, 17, 109], public health [91], natural disasters [104], and epidemics [27]. A nice example of the use of the public mood could be seen during the Olympic games in London, 2012.

Daley Thompson, a famous athlete, created a light show that was exhibited on the London eye. A real time sentiment analyser classified the tweets about the Olympic games into positive and negative categories, and lit the London eye according to the percentage of positive tweets.

Another illustrative example of the importance and the wide spread usage of sentiment analysis in the context of financial news is the dramatic drop in Dow Jones Industrial Index in April 2013. At that time, the twitter account of Associated Press was hacked, publishing some information about an attack in the White House in Washington, and an injury of Barack Obama. The index responded to the news with an immediate drop, shown in Fig. 1.1. The hack was clarified shortly after the incident, and the index recovered directly. This story shows how the sentiment analysis field is of a growing importance.



AP The Associated Press @AP

Breaking: Two Explosions in the White House and Barack Obama is injured

1,900 RETWEETS 83 FAVORITES

1:07 PM - 23 Apr 13

Figure 1.1: The left hand side figure shows the tweet from the hacked AP Twitter account that announced the attack on the white house the injury of Barack Obama. The right hand side figure shows the crash of DJIA after the hacking of AP. Source: FactSet, MarketWatch

1.1 Background and Problem definition

In computational linguistics, sentiment analysis is considered to be a classification problem. It involves natural language processing (NLP) on many levels, and inherits its challenges. There exists a wide variety of applications that could benefit from its results, such as news analytics, marketing, question answering, knowledge bases and so on. The challenge of this field is to improve the machine's ability to understand texts in the same way as human readers are able to. Taking advantages from the huge amount of opinions expressed on the internet especially from social media blogs is vital for many companies and institutions, whether it is in terms of product feedback, public mood, or investor opinions.

The present thesis searches into different possibilities to improve sentiment classification performance. To address this problem, three different key issues are investigated. The first issue is to improve sentiment classification through text pre-processing. The second issue is to improve it through utilising text properties. The third issue is to improve it through inferring sentiment from one domain to another. These issues are explained in the following.

1.1.1 Text Pre-Processing: Movie Reviews

Online texts contain lots of noise that can cause misleading results in the sentiment classification process, such as html tags, advertisements, hyperlinks, stop words and words that bear no effect on the text orientation. In the analysis, each text is represented by a vector where each word is an entry. Thus, each word is a one dimension in the vector space. Therefore, many text documents are of high dimensionality and this makes the task of the classifier more complex. That is because higher dimensionality leads to higher sparsity which makes it harder to find similar properties among the classification targets.

These issues can be resolved by using natural language processing techniques, the so called text pre-processing. Two aspects are investigated in parts of the present thesis, these are: to what extent text pre-processing helps to improve the classification performance and what types of NLP techniques can be implemented. Text pre-processing and dimensionality reduction are well addressed in the fields of information retrieval and text mining. Yet, there are few studies that address in particular the importance of text pre-processing in sentiment analysis.

To answer these questions, a computational framework to carry out data pre-processing on movie reviews data is suggested in this thesis, and its effect on the classification outcome is investigated. This data set is widely used in sentiment analysis [1, 90] which allows a direct comparison to previous approaches. Furthermore, the analysis is conducted on a document level. Thus, a single sentiment score is computed to assess overall sentiment of each text in the data set. Despite the critiques to this approach that will be explored in the next chapter, it is argued in this thesis that this analysis is suitable for reviews. That is because a final statement about the product in question is needed even if its review contains several different opinions about it. Out of the three different approaches to sentiment analysis that will be discussed in Chapter 2 (machine learning, lexical, and linguistic), a machine learning approach is taken in this thesis.

1.1.2 Text Properties: Financial News

Two properties of financial news can be used to create a model for sentiment analysis. The first property is that the financial news contains announced events about publicly listed companies. Those events have a direct effect on the corresponding companies stocks. This relation could be employed to build a model that predicts the sentiment based on the type of the announced event.

Generally, opinions can be inspected from two perspectives: opinion holder, and opinion reader. The scope of sentiment analysis in reviews is the opinion of customers who write online reviews, in other words the opinion holders. The reader opinion reflects what the reader thinks of the review. The collection of the reader views is not an easy task. Some web pages allow readers to express whether they have found the reviews in particularly useful or not, either by a direct question to the reader (was this review helpful?), or by adding a like/dislike button. In this case, the reader opinion is a reflection on how they found the products they purchased and how helpful the reviews were. One could use the products sales figures as the reader sentiment and investigate the correlations to the reviews/holder sentiment to study how sales corresponds to positive or negative product reviews. No studies are conducted to investigate the reader opinion [71].

The present thesis sheds light on this issue. The question that is raised here is how the reader sentiment can be captured. Here comes the second property of financial news. It is argued in this thesis that financial news allows to deduct the news reader sentiment through stock market returns. The reader in this respect is the investor in the market. Accordingly, the correlation between investors' sentiment and stock price fluctuations is under investigation. How can the investor opinion be translated into a signal for buying or selling? Understanding the investors' opinions and determining their sentiment is part of the field of sentiment analysis. A method to evaluate the investor's opinions and use them to label news items automatically is proposed in this thesis. The data is collected from internet resources. The collection process of financial data will be explained in details in Chapter 4, as well as the computation and evaluation of two novel models to predict the sentiment of financial news.

1.1.3 CROSS DOMAIN SENTIMENT: VARIOUS DOMAINS

After the investigation of financial news data, it can be noticed that this class covers a wide variety of different domains. Some news cover companies law suits which are in the legal domain while some others cover the annual profits or earning announcements which are in the financial accounting domain. Merger or acquisitions can be legal or financial issues. The interesting question is: does this domain variety affect the classification performance of the sentiment classifier?

Before this question is answered, it should be considered that one preeminent essence of sentiment analysis is its domain dependency. That means the same word or sentence can express different sentiments in two different domains. An illustrative example is the sentence *go read the book* [89]. While this sentence is positive in a book domain, it is negative in a movie domain. This dependency makes it complex to transfer sentimental knowledge from one domain to another. A classifier that is trained on one domain with high predictability, performs significantly worse in another domain [86, 13]. The question is how to transfer the sentimental knowledge across domains and allow for a better performance in the secondary domains? Most of the research focuses on domain independent words to bridge between two domains [93, 86, 12]. A method to detect domain dependent words and deduce independent words is suggested in this thesis. The list of dependent and independent words are utilised by several suggested models to classify sentiment over different domains. Results from experimental work on different cross domain algorithms are reported.

1.2 AIM AND OBJECTIVES

The main aim of this thesis is to explore key ways of improving sentiment classification performance. To achieve this, there are three distinctive objectives.

The first objective aims to improve sentiment prediction through text pre-processing. A wide variety of pre-processing methods is presented and an appropriate feature selection method is selected for the analysis. Document level sentiment classification is performed along with the focus on products reviews and the use of movie reviews as an example.

The second objective intends to improve sentiment classification through delving into various text properties. The example here is financial news that has two properties. Firstly, the financial news contains announcements of financial events that could be utilised in the sentiment prediction. A model that employs news events in sentiment classification is proposed. Secondly, financial news allows for capturing the investors (reader) opinions through stock market returns. It is argued in this thesis that in some tasks such as financial forecasting, it is the sentiment expressed in the responses of content readers (for instance, through trading behaviour) that may be more useful as a means of creating predictive models. A new model that is built to predict financial news sentiment based on a novel method to capture reader sentiment is presented.

Furthermore, the financial news covers a wide variety of different domains such as economics, accounting, law, etc. Therefore, the third objective aims to improve sentiment classification through investigating the case of cross-domain sentiment analysis. A method for selecting domain dependent and independent words is proposed, and a new model for cross domain sentiment analysis is evaluated against other approaches.

1.3 Contributions

The main contributions of the present thesis are as follows. In Chapter 3, the impact of text pre-processing in the sentiment analysis domain is investigated in depth. A two level combination of text pre-processing methods is suggested to carry out a sentiment classification process. The sentiment prediction results have shown significant improvements in the classification performance. Furthermore, new extensive experimental results that demonstrate the effect of the feature sets sizes on the classification performance are reported. They also illustrate the relationship between negative and positive words occurrences in a certain document and the assigned polarity to this document. The results show that negative words has a bigger impact on the classification performance, and that less negative words are needed to get true negative prediction with high confidence. On the other hand, more positive words are needed to make a true positive prediction with high confidence. Moreover, the random selection of training sets reflects on a high dispersion in classifications accuracies among several runs. This suggests a new way of reporting sentiment analysis results in a way that shows this range of accuracies.

In Chapter 4, a new model to predict the sentiment of financial news based on events occurrences is suggested. The model tries to associate certain events with certain sentiment labels, and assigns those labels to any piece of news that carries any of the events under investigation. The reported results suggests that this model cannot be ruled out, and that bigger data sets are needed to test its reliability.

On a different scale, the problem of capturing the reader sentiment is approached for the first time in this thesis. A new method to capture such sentiment (RSInd) is suggested. The method utilises abnormal returns to associate news items with certain positive or negative labels. RSInd is used to build a new model to predict the sentiment of financial news. The prediction results show that this model can be used successfully as a stocks trading strategy.

In Chapter 5, a wide variety of different methods in cross-domain sentiment analysis are explored. A new method of identifying domain dependent and independent methods is suggested. The method identifies the threshold that splits domain dependent from independent words based on words frequencies and mutual information scores. A new method for cross domain is proposed based on relevance networks and compared to other methods. The results of this method still can be improved and tested on bigger data sets. Furthermore, new extensive experimental work is reported in this chapter for different models to predict cross domain sentiment. The models are compared and evaluated against each other.

1.4 Thesis outline

Chapter 1 opens up with the focus of the current thesis and sheds light on the field, different approaches and questions raised in sentiment analysis and how these are approached in this thesis.

Chapter 2 delves into various methods that are used in sentiment analysis. Natural language processing techniques, including feature selection which is used in the pre-processing of texts are investigated. Furthermore, the machine learning approach taken in the thesis is explored.

In Chapter 3, different text pre-processing methods are combined to classify the sentiment of movies reviews. Experimental results are reported which demonstrate that with appropriate feature selection and representation, sentiment analysis accuracies using support vector machines (SVM) can be significantly improved. The level of achieved accuracy is shown to be comparable to the ones achieved in topic categorisation although sentiment analysis is considered to be a much harder problem in the literature. The relationship between the documents size and classification performance is also investigated where the results show that the classification performance can be different among the two categories based on the number of features in each document.

In Chapter 4, a different type of online texts is inspected: financial news about publicly traded companies. Two different models are proposed to detect the sentiment of financial news: event-based and reader based models. The proposed method in Chapter 3 is utilised to classify the sentiment of financial news. Automated labelling based on the investor (reader) sentiment is used to classify news for training purposes. This means that texts linked to positive stock market returns are categorised as positive. If the returns are negative, then the text is categorised as negative. An amended automated labelling of financial news is proposed. It employs abnormal returns instead of plain returns that have been used previously in the classification process. The models are then validated against other approaches.

In Chapter 5, two different methods to identify domain dependent and independent are proposed and combined together. One method considers the feature frequency in each category in relation to its significance. The other method considers the features mutual dependency on the document category. Furthermore, those methods are utilised in three different models to transfer sentimental knowledge from one domain to another. The models are validated against each other and against the proposed methodology from Chapter 3. Finally, an attempt to build a lexical classifier is presented and the results are reported.

Concluding remarks and proposals for future work are given in Chapter 6.

CHAPTER 2

BACKGROUND

Sentiment analysis is the “field of study that analyses people’s opinions, sentiments, evaluations, appraisals, attitudes, and emotions towards entities such as products, services, organisations, individuals, issues, events, topics, and their attributes” [71].

Another term, which expresses sentiment analysis, is opinion mining. Before discussing opinion sentiment, one should address the definition of opinions. Liu [70] defines opinions as “subjective expressions that describe people’s sentiment, appraisals or feelings towards entities, events and their properties”. He distinguishes between a direct opinion which gives a positive or negative statement about an object, and a comparative opinion which implies positivity or negativity by comparing the object to other objects.

Kim and Hovy [57] define the opinion as a combination of four factors: topic, holder, claim, and sentiment, while Liu [70] defines five factors of a direct opinion: object, features, orientation, holder and time. In the first case the opinion holder has a claim about a certain topic and associates this claim with a certain sentiment. In the second case, the opinion holder finds some features of a certain object, and associates a certain orientation about them in a specific time.

In computational linguistics, sentiment analysis is approached as a classification problem; texts are classified into positive or negative classes. If the text is lacking an opinion, it is then labelled as neutral. In many cases, the classification appears as a multi-categorisation task instead of a binary one, such as in a ranking basis sentiment, where a multi points scale is used to rank texts [89].

Abbasi, Chen, and Salem [1] summarise sentiment classification characteristics. These contain tasks, features, techniques and applications. The tasks are classifying the text into classes: negative vs. positive, subjective vs. objective, and defining the level of the text classification as phrase, sentence or document levels. The features are one of four categories: syntactic, semantic, link based and stylistic. The techniques that are used in the classification are either machine learning based, link analysis[1] or similarity scores bases[2].

Little research had been conducted to explore opinions before the year 2000. According to Liu [71] the term sentiment analysis first appeared in [32] although there were previous studies that explored texts sentiment [30, 89].

The difficulty of classifying a text as positive or negative depends on the type of the text: whether it expresses a fact or an opinion. Generally, classification of opinionated text is more difficult than it is for factual ones. That is because opinionated text could carry some speech informality that makes the opinion detection harder, such as sarcasm, subjective language, emoticons, and so on. Liu [70] claims that much of the research in textual sentiment classification was focused on factual texts due to their availability compared to opinionated texts which have recently become widely available in the social media blogs on the internet, and that happened to be called “user-generated content”. It is almost impossible for humans, or companies to retrieve the latest trends and summarise the state or general opinion about products due to the diversity and sheer size of the social media data. This creates the need of automated opinion extraction and mining and summarising methods.

[1] “message send/reply patterns and citation analysis” See Abbasi et al. 2008

[2] “phrase pattern matching , frequency count” see Abbasi et al. 2008

2.1 Approaches toward Sentiment Analysis

2.1.1 Sentiment Classification Methods

There exist three different methods in sentiment analysis: machine learning based methods, lexical based methods and linguistic analysis [119]. Machine learning methods for sentiment classification are based on models that are calibrated to categorised data. This data is called training data. The calibrated model or machine can then be used to categorise new data, much like a parametrised equation can be used to predict the value of the response variable in regression analysis. The training is based on features that have an effect on the data polarity, and they are chosen using feature selection methods that will be explored later in this chapter.

Lexical based approach depends on constructing a *Lexicon*, which is a "structure that keeps track of words and possibly information about them" where the words are referred to as "lexical items" [21] . Once the lexicon is constructed, the overall polarity of the text is then found by a possibly weighted count of those lexical items [53, 35, 4, 77, 81]. One example of constructing a lexicon is to choose the so called opinion-bearing or polar words. Then those words are split into two categories based on their polarity, and used then to build the lexicon [70].

Lastly, the linguistic approach uses the syntactic characteristics of the words, phrases, negations, and the structure of the text to estimate the text orientation. This approach is usually combined with a lexicon based method [114, 119, 122]. One of the methods used in the linguistic approach is based on Parts-Of-Speech (POS). POS defines the syntactic patterns or categories of the words [21]. Various patterns of POS can be used as phrases to be selected from the text. Those patterns indicate either a specific sentiment or a specific topic. It might contain adjectives or adverbs or any other part of speech [61]. To define the patterns, n-grams are used. An n-gram is a sequence of n words from a given sequence of speech. One can use unigrams, bigrams, trigrams, and n-gram for more than three words. Suppose n-grams are used for patterns to search for causal sentences. A unigram could be *because* or *since*, a trigram could be *as it is* or *that is why*.

The three different approaches can be used separately or combined together. For example, machine learning and linguistic approaches can be combined, so that the selected features for training are only features of one POS kind. A lexical based analysis can be combined with a linguistic approach, so the lexicon is built for example from the adjectives that appears in a text or in a specific domain. Those adjectives might be classified in a lexicon as positive or negative. For instance,'beautiful' and 'ugly'. This does not imply to ignore the importance of other parts of speech like verbs or nouns, since some of them express very strong sentiment, such as the verb *hate* [89].

2.1.2 WHOSE VIEW?

This section approaches the sentiment problem from a different perspective. Whose sentiment is investigated? As mentioned earlier, the opinion holder is a main factor in the opinion construction. The opinion holder could be the opinion writer, or a person that the writer is attempting to transfer their opinion. Most of the research in sentiment classification analyses the opinion from the opinion holder point of view, but there is as well the readers view and their reaction towards the expressed opinion [71]. The opinion reader could stand in the same or in a different position from the opinion holder. Consider for example news about housing prices fell down. Economically, this is not a good piece of news. It is also bad news for sellers while it is good news for buyers [71]. It is not an easy task to gather information about how the opinions reflect on the readers and their sentiment. However, this is feasible in some cases like financial news about publicly listed companies where the reader sentiment reflects in stock market returns. This feasibility is utilised in this research as financial news sentiment will be investigated from the reader point of view in

Chapter 4.

2.2 Depth of Analysis

Sentiment analysis can be conducted on different levels of granularity: document, sentence, word or phrase, aspect or user levels. The aspects of these levels are explained in the following paragraphs.

2.2.1 Document Level

Document level sentiment analysis targets the whole document and assigns an overall sentiment to it, assuming that the document expresses a single sentiment [29, 90, 73]. This assumption is criticised as unrealistic because a text could hold more than one opinion and hence, the analysis should target finer levels [71]. However, this assumption holds for some branches like reviews, where a final statement about the product is required which is a weighted conclusion arising from different aspects even if the review carries different opinions. Another case where this assumption is valid is in financial news where the news that carries a positive or negative sentiment reflects in a buy or sell signal. In this light, two types of texts, movie reviews and financial news are considered in this thesis.

2.2.2 Sentence Level

Sentence level is a finer level of analysis that inspects sentences which express a single opinion and try to define its orientation [137, 114, 97, 112]. The assumption here is that each sentence carries a single sentiment. This assumption does not hold for all sentences in a text. In fact, many sentences do not carry a specific sentiment. It is important to distinguish subjective from non-subjective sentences, because non-subjective sentences add no information to the classifier [58].

Subjective statements are those that express an opinion. Subjectivity detection is the separation of sentences that contain opinions from those that contain facts [129]. Even in opinionated texts, tagging subjectivity prevents misleading and irrelevant sentences from affecting the sentiment classifier [117]. Hence, subjectivity classification is used to improve the performance of sentence level sentiment classifiers [87, 137, 57, 70, 80]. Sentence level is preferred over document level when different opinions in one document are needed to be captured.

Texts consist of various types of sentences, each of those has several particular characteristics that make it possible to treat it differently and allow different types of special classification for them, for example, conditional or comparative sentences. It is claimed that there is not one classification strategy that fits all the types of sentences or even a whole text. Therefore, combining different strategies based on different types of sentences improves the classification accuracy [84]. Generally, in an opinionated text or a review the opinion holder is likely to express an overall positive or negative opinion. However, the review author might mention good and bad features about the object [70].

2.2.3 Phrase and Aspect Level

Phrase or word level analysis investigates the polarity of texts on a finer level: the phrase level. This involves distinguishing polar phrases and then defining their sentiment [131, 115]. In finance, word level sentiment is used to detect polar words and measure their relationships to other variables like firm earnings or stock prices [118].

Recently, many statistical models have been built for a deeper analysis of product reviews, that is, mining the customers opinions about certain product features [47, 48]. This is commonly referred to as aspect level sentiment analysis. It is the process of extracting relevant aspects of the reviewed product and determining the sentiment of the corresponding opinion about them [95, 24, 11]. The assumption here is that all opinions are generally directed at a specific topic/object which none of the above frames targets accurately and consistently [71]. For example, in movie reviews, extracted aspects could be: music, actors, or lights. When the customers are writing about a movie, they express their opinions about these aspects like what they think of the actors or the music choice.

2.2.4 User Level

In addition to the previous levels of analysis, some studies carry out the analysis on a user level that looks into users' networks and predicts users sentiment based on the sentiment of neighbouring users [28, 77, 98, 113].

In addition, a number of studies use joint models that combine two or three different levels, where the knowledge about one level, say documents, helps predict the sentiment of another level, say sentences [75].

2.3 Sentiment Analysis for Reviews

Sentiment analysis in reviews is the process of exploring product reviews on the internet to determine the overall opinion or emotions about a product. Reviews represent the so called user-generated content, and this is of growing attention and a rich resource for marketing teams, sociologists and psychologists and others who might be concerned with opinions, views, public mood and general or personal attitudes [117] .

The huge number of reviews on the web represent the current form of user's feedback. Deciding about the sentiment of opinion in a review is a challenging problem due to several factors. One issue is the subjectivity in the texts and the need to identify opinion bearing from non-opinionated sentences. Another issue that makes it hard to classify reviews is the so called “thwarted expectation” which means that the writer writes many sentences in one direction which can be understood as positive, and then concludes with one negative sentence that reverses the meaning of the entire text [90]. This is why there should be better methods for feature selection. Turney [122] concludes that movie reviews are hard to classify because the overall opinion represented in the review is not necessarily the sum of all of the opinions mentioned in the text. Many studies have explored various methods in analysing product review sentiment, some of which use machine learning approaches [90, 136, 55, 59], some use lexical methods [122, 95, 32], and some use linguistic approaches [83] or combine different approaches [62, 135].

The fact that there are a large number of online reviews and that some of these contain only a small fraction of sentences that express an opinion, makes it harder for customers and companies to know the overall opinion about a product and to establish an opinion about it. For this reason, opinion summarising is proposed. Opinion summarising could be conducted by rewriting the original text and focusing on the main point. It also can use aspect level sentiment analysis, and select the main features in the texts, their aspects, and the corresponding opinions to generate a “feature based review summary” [47, 48]. Examples can be found in [63, 47, 48, 11, 68].

In this thesis, movie reviews are investigated, using one of the most famous data sets in sentiment analysis [88]. The problem is approached using machine learning methodology, and the opinion holder perspective.

2.4 SENTIMENT ANALYSIS FOR FINANCIAL NEWS

Stock market prices and their fluctuations are analysed on a large scale everyday. Day after day, more and more techniques are invented to yield a better understanding of the price time series in order to achieve higher profits. Asymmetric information creates better chances to gain payoffs before the entire market knows about them. This need for superior knowledge endorses the need of automation and the ability to quickly retrieve and analyse new coming information. This process is known as “News Analytics”.

News analytics in finance is the operation of processing the news using IT tools and algorithms, to detect information that gives a wider understanding about the companies and a base for the decision making process of the investors. The events which occur when the news are published have the most focus. When did this event happen, where did it happen, and who was the publisher, which market or which company is affected, and most importantly is it good or bad. In the literature, the information is defined as: news intensity, relevance, sentiment scores, novelty and the type of items [67]. News intensity is the number of news articles in a pre-specified period, while the relevance determines how closely a story in the news is related to a stock or a company. As for novelty, the newest story has the strongest effect, and the novelty scale of a hundred percent indicates that the story has just recently been published. The sentiment score reveals whether the story is bad or good [79]. This sentiment score is the main subject of Chapter 4 in this thesis.

The information gained from the news text can be quantified and used to measure correlation with liquidity, stocks trading volumes, stocks returns, volatility, risk estimation and other economic variables [60, 10, 78]. However, a new trend in algorithmic trading is to incorporate techniques to measure the overall sentiment of investors because their decisions must be driving to some extent the changes in the stocks prices. Investors’ opinions can be crawled from message boards on the internet such as Seeking-Alpha or any other financial forums or message boards on the internet [6, 31]. Another current trend is the investigation of the relationship between the general mood of the public and the stock market direction [18].

Corresponding to this thesis interest in investor’s opinion, two questions could be raised. How is news affecting the investors? What is the investor sentiment corresponding to news? Blood and Phillips [14] conclude that the consumer sentiment is affected by the news headlines about the economic recession over the period of study, while it is not influenced by the state of the economy. Behavioural economics show how emotions can drive the decision making process. Bollen, Pepe, and Mao [18] show that stock price predictions can be improved by adding the factor of general mood to the other factors of prediction.

Next to the common approaches towards sentiment analysis, there exists an additional approach for financial news. This approach relates to the specific properties of financial news. Two properties are identified in this thesis. Financial news contains in many cases announced events about certain companies. The other property is that financial news allows for capturing the reader sentiment throughout market returns. As news analytics concerns the announced events in the news[3], financial and economic experts are able to identify the event in a piece of news and then use economic knowledge to decide about its sentiment and how strong it is, that is the related degree of positivity or negativity. This method of measuring sentiment scores is used in the world leader companies of news analytics [43].

In this thesis, financial news are inspected. The problem is approached using event occurrences, machine reading methodologies, and the opinion reader perspective.

2.5 CROSS DOMAIN SENTIMENT ANALYSIS

Financial news covers a wide range of domains. The news about publicly listed companies can be related to legal, accounting, economic, marketing, and many other domains. The fact that sentiment is domain dependent, and that words that are used to describe a good sentiment in one domain are different from those used

[3] Event studies, measure the effect of the event occurrence on the stock returns. It was introduced by Fama et al. in 1969, to provide a proof of the stock returns response to the information (see Fama, 1998).

in another domain, makes it more complex for a classifier to select features that are relevant to the sentiment in all covered domains by the news. That creates the need for a cross domain classifier that is able to transfer the sentimental knowledge from one domain to another.

At the heart of cross-domain sentiment analysis is a classifier that is trained on one domain (source domain) and then used to predict another domain, the target domain. For example, assume there exists a set of sentiment-labelled documents in movie reviews. How could a classifier be trained on this data set and then used to predict the sentiment of book reviews or electronics reviews? Words that are expressing the sentiment in one domain are called domain dependent words. Those words need to be taken in consideration within the classification process to minimise the decrease in prediction accuracies across the various domains.

There are several studies that explore cross domain sentiment analysis, using various approaches and differing algorithms. To find connections between texts in different domains, domain independent words are selected. They are words that appear in all domains and have a similar impact on the texts orientation [86]. Most of the research aims to find domain independent words to bridge between two different domains by looking into their co-occurrences with domain dependent words. Optimally, those domain independent words have high rankings, or are of significant impact on the text orientation in both domains [13, 12, 7, 116, 134]. Consider for instance the sentence *The movie is very good. I recommend it* from the movie domain, and the sentence *The picture quality is very good*. The feature “very-good” is an example of an independent feature that can be used to bridge between the two sentences in both domains in order to transfer the sentiment from one domain to the other. In other studies, independent words are used in a graph based models for sentiment predictions [86, 132, 94, 93]. Others use topic modelling to identify the opinion topic and use it to identify similarities to bridge both domains and allow to transfer sentiment [45, 41].

Some domain dependent words are non-informative from a sentiment classification perspective as they bear no effect on the text polarity. For example, the words book, teacher, classroom in the education domain, and the words movie, actress, scene in the movie domain. Those words are treated like stop words. Removing them from the text helps to reduce the text dimensionality and hence to improve the classification performance. To distinguish them from domain dependent words that bear an effect on the text polarity, they are called domain specific words in the course of this thesis. In most cases domain specific words are chosen based on a prior knowledge of the field which makes them not valid for other fields. Xia and Zong [133] explored several part of speech tags that turned out to be domain specific and built a model to integrate POS tags to improve the classification accuracy. More in-depth-analysis about domain dependent and independent words, and experimental work on cross domain sentiment analysis are presented in Chapter 5.

2.6 PRE-PROCESSING TEXTUAL DATA

Pre-processing the data is the process of cleaning and preparing the text for the classification process. The necessity for this step lies in the fact that online texts contains usually noise and uninformative parts such as HTML tags, scripts and advertisements. In addition, on words level, many words in the text do not have an impact on the general orientation of it. Since each word in the text is treated as one dimension, keeping irrelevant words increases the dimensionality of the problem and hence makes the classification more difficult [100]. The difficulties do not only manifest themselves in the robustness of the analysis, but also in the computational complexity of the classification process [22]. The entire pre-processing procedure involves several steps: online text cleaning, white space removal, abbreviation expansion, stemming, stop words removal, negation handling and feature selection.

All of the steps but the last one are called *transformations*, while the last step is called *filtering* [38]. Pre-processing and its major impact on text classification accuracy are discussed further in Chapters 3 and 4, respectively.

2.6.1 TRANSFORMATIONS

HTML CLEANUP

Web-pages contain in addition to the main texts advertisements, and other irrelevant information such as HTML tags (for example <p>,<br>). These are organised in different object elements, i.e. so called <div> tags. To avoid efficiency problems that arise from that irrelevant information, the text should be cleaned from them to retain only the information of interest. There are many ways to extract the relevant news texts from the HTML source code. For example, one can use “HTML Cleanup” which was used in [29], or the document object model(DOM), or the Apache library HTMLUnit which can parse the HTML specific features from texts, and arrange them in an object-based tree structure to be distinguished and separated from each other. This structure allows then to extract the core text of interest.

EXPANDING ABBREVIATION

Next to computer language specific pollutions of the text that have been discussed in the previous paragraph, abbreviations can create noise in the course of the analysis. This problem is solved by abbreviation expansion. For instance, *they’re* is substituted by *they are*, *hasn’t* is substituted by *has not*, and so on [29]. This helps on one hand to get the correct frequency of the words and then the correct dimension of the text. On the other hand, expanding the negation part as in *don’t like* → *do not like* helps to detect and tag the negation in the sentences and facilitate their detection. Negation cases are of great importance in sentiment analysis as they reverse the sentence sentiment. More about negation handling will be explained later.

White Space and Stopwords Removal

Some parts of the text might contain two white spaces especially after the removal of the HTML tags. White space removal is the process of removing one of those spaces for each occurrence of two spaces. Stopwords are words which have no discriminant value in the text, or do not add any information to the general orientation of the text in terms of sentiment classification. Moreover, their existence causes less accurate results and longer processing time due to the increase in the text's dimensionality without additional information. Makrehchi and Kamel [74] divide the stopwords into two groups, general stopwords and domain specific stopwords. General stopwords are either standard and the are available in public domain or non-standard and they are generated from the systems of text categorisation or information retrieval.

Domain specific stopwords are words that are used in a certain domain. The stopword in one domain can be a keyword in another one. For example, the word *learning* is a stopword in any education domain while it is a keyword in the computer science domain. Although domain specific words are essential in text categorisation or topic categorisation, they are of little importance in sentiment analysis as they do not have an impact on the text orientation. Therefore they are usually removed among general stopwords for dimensionality reduction. However, identifying domain specific words could be useful in cross domain sentiment analysis as will be explained in Chapter 5.

There exist different methods of stopwords removal. One of them uses a list of stopwords, called a stoplist. The stoplist contains words that are considered to be non-informative generally. Such stoplists are publicly available. In the literature, the Rijsbergen stoplist [124] is one of the widely used stoplists in natural language processing. Still, stoplists are continuously outdated after their publication, and that is because of the change in words usage over time due to social factors and changes in technology [74]. Therefore they should be always scrutinised and updated. Other ways of constructing stoplists is based on the frequencies of the words in a text. Words with high frequencies are treated as stopwords. As for domain specific stopwords, stoplists are usually constructed from a dictionary or a corpus for the domain in question [74, 15, 16]. Alternative approaches utilise statistical methods such as mutual information [108, 92, 72]. Both frequencies and mutual information are utilised in this thesis.

STEMMING

Stemming is the process of deleting the suffix of the word and putting it into its basis or fundamentals. If a text contains the words, admire, admired and admiring, they should not be treated as different words especially in a sentiment classifier where they have the same meaning and the same polarity. Stemming them will return them to “admire” and then the word’s frequency will be 3, and that is instead of three words with a frequency 1. The importance of this step lies in decreasing the dimensionality of the text[38]. For classifiers like support vector machine, each word is considered to be a vector in its dimension. Hence the number of different words in a text represents the number of dimensions. Stemming allows considering the word and all of its derivatives as one word. Hence, stemming helps not only to reduce dimensionality but also to correctly identify words weights and importance in a text through their frequencies.

The Porter stemming algorithm or the suffix stripping algorithm is a widely used algorithm for stemming and was introduced by Porter in 1979 in a project of Information Retrieval (IR)[125]. Porter’s argument is that stemming helps to improve the performance of information retrieval. The IR system has a collection of documents which in turn contain words most importantly in the headlines and the abstract. Porter confirms that the document is represented by a vector of terms and that the IR system will perform better if words like “connect, connected, connecting, connections” will be grouped into a single term “connect”. As a result of applying his stemmer the size of the new vocabulary collection was as much as two thirds of the one before the algorithm was applied [96]. In this thesis, the stemming effect on sentiment classification will be addressed in Chapter 3.

HANDLING NEGATION

Handling negation is of high importance in sentiment analysis due to the fact that one negation word would change the polarity of the sentence from one side to the other. An example is the two very similar sentences that have opposite sentiments, *I like this book, I don’t like this book* [89]. There exits different types of negations. The sentence may contain a direct negation such that the negation word and the negated words are neighbours, take for instance “not nice”. The sentence may also contain a long distance negation where the negation and negated words are separated such as ”not very interesting, does not have good music”. The negation could be for the subject (e.g., “no one liked it”), or the verb (e.g., “did not succeed”), or adjective/adverbs phrases (e.g., “not really interesting”). In addition to those different types, in some cases, negation words do not reverse the sentiment of the sentence. For example, in the sentence “Not only the actors choice attracted me but also the music”, the negation does not reverse the sentiment, it enhances it. Thus, negation handling is important in sentiment analysis.

Negation can be controlled in different ways. Some studies use a linguistic approach to handle the negation by composing a negation phrase and treating it as a unigram [83, 32]. The negation phrase could be the negation word with corresponding negated word, for instance, “not good” will be “good_NOT” [31]. It could also be a phrase that contains the negation word and the all the words that occur after the negation until the first punctuation appears [90]. Others distinguish between the different types of negation that are mentioned previously and tag the negation on a phrase level by combining different structures of negation phrases [131, 26].

Narayanan, Liu, and Choudhary [84] use two strategies: the first one is to tag the negation word as a feature, and the second is to reverse the sentiment after finding the sentiment of the sentence. They report that the first approach is more accurate in terms of correct classification.

2.6.2 FILTERING

FEATURES SELECTION

Features in the context of opinion mining are the words, terms or phrases that strongly express the opinion as negative or positive. This means that they have a higher impact on the orientation of a text than other words in the same texts. As many of other pre-processing techniques, the main reason to pick up these features lies in reducing the dimensionality of a text to achieve higher classification accuracies. Features could be explicit or implicit. A feature is considered to be explicit if it appears in a text, whereas it is considered to be implicit if it does not appear but it is implied in the meaning [70]. For example, the sentence *The picture quality is good* has an explicit feature "picture quality", while the sentence *this camera is too heavy* has an implicit feature "weight". The present thesis addresses only explicit features. Features can take the form of unigrams, bigrams or n-grams subject to requirements. While some studies find that better polarity classification can be achieved by using bigrams and trigrams instead of unigrams [32], others find that unigrams outperform bigrams in sentiment classification [90].

There are several methods that are used to select the most important features in a text. Some of them are discussed in the following. Several methods of selecting features from a text consider the syntactic position of the word such as adjectives, adverbs, or verbs [25]. Na et al. [83] could improve the accuracy of their classification when they limited the features to adjectives, verbs and adverbs. This matches the hypothesis that negative and positive statements are mostly expressed by adjectives, adverbs and verbs [83, 31]. One possible implementation of this is to keep only those words and remove all other words, or they can be given more weight than other words within the texts. In aspect level sentiment analysis, syntactic based feature selection is used to select the nouns that the opinion is expressed about and then attach them to the closest adjectives that they relate to [36]. For example, in the sentence *The picture quality is good*, the selected noun is the bigram “picture quality”, and it is then attached to the corresponding adjective “good”.

Another way to select features is to assign weights to them and then define a certain threshold for the targeted features. The most common approaches to weigh features in a document are:

- Feature Frequency (FF): the weight of the feature is its number of occurrences in the document.
- Feature Presence (FP): the weight is binary. It takes the value 0 or 1 based on the feature absence or presence in the document.
- Term Frequency Inverse Document Frequency (TFIDF). It is calculated for each feature by the formula

  $$TFIDF = FF \cdot \ln\left(\frac{N}{DF}\right), \tag{2.6.1}$$

  where FF is the feature frequency, N is the number of documents, and DF is the number of documents that contains this feature [83].

O’ Keefe and Koprinska [85] compare the impact of the different weighting methods using SVM and Naive Bayes classifiers. They conclude that the FP method is the best among the others. This result confirms the findings of [90] as well as the findings of Chapter 3 in this thesis, but it is conflicted with the findings of [83] which conclude that TFIDF is slightly more effective than FF and FP.

Information Gain is a popular method for feature selection in high dimensional texts. This method measures the effect that each feature has in the text category. It measures the relevance for each of the features in the text, and then select the words with the highest relevance [65]. Information gain in text mining refers to the additional information that a term or a word in a text contains. It is derived from entropy which is the amount of information in a text. The higher the information gain of a term is, the higher is its impact on the text orientation [139].

Another popular method for feature selection is the χ^2 Statistic [69]. It is a statistical analysis method that is used in text categorisation to measure the dependency between a word and the category of the document it is mentioned in. If the word is frequent in many categories, χ^2 value is low, while if the word is frequent in few categories then χ^2 value is high. This method has shown good results in feature selection [123, 76].

The null hypothesis of the χ^2 test is that the feature is independent from the category (pos/neg in this case). If the p-value of the corresponding test-statistic is smaller than 0.05 then the null hypothesis can be rejected and the category is dependent on the feature. The statistic can be computed with [99]

$$\chi^2 = \sum_{i=1}^{n} \frac{(O_i - E_i)^2}{E_i}, \tag{2.6.2}$$

where O_i is the observed frequency of feature i, and E_i is the expected frequency of feature i under the assumption of the null hypothesis that the feature is independent from the category.

Another approach for feature selection is a correlation based approach that searches for appropriate features by investigating their correlation to the proposed categories and selects the features with stronger correlations [138, 44].

In addition to the previously mentioned methods for feature selection, there are many other methods that are used in the literature such as principle component analysis (PCA), Genetic Algorithms (GA), classification and regression trees (CART), Ng-Goh-Low-Leong (NGL) coefficient, Galavotti-Sebastiani-Simi (GSS) coefficient, Odds Ratio, Fisher Criterion, the GU metric [121, 123].

2.7 DOCUMENT SENTIMENT CLASSIFICATION

The sentiment of a text or an opinion is considered as a classification problem, since texts are assigned one of several categories. Generally, there are two classes: positive and negative (bullish or bearish in financial jargon), hence it is a binary classification problem. In some cases, an additional class “neutral” is added to the positive and negative classes, and it expresses a lack of opinion. In other cases, classification is based on a ranking factor such as the star based ranking. Then the problem becomes a multi-class categorisation task [89].

There exists plenty of standard classification algorithms that are used in text mining and machine learning. Examples are Bayes Classifier, Support Vector Machines (SVM), Word Count Classifier, Vector Distance Classifier, Discriminant Based Classifier, Bayesian Classifier, and Adjective-Adverbs Classifier. Some of these classifiers are language dependent and some are not, and they all can be used for a large set of data [31, 29]. Language dependent classification means that the classification process is based on the linguistic characteristic of the text such as the presence of some POS patterns like adjectives or adverbs. The classifier that is used in the present thesis is SVM.

SVM [126] has become a very auspicious method of classification and regression for linear and non-linear problems [66, 65]. It was first successfully used for text classification by Joachims [51]. SVM has a high performance and is widely used in text categorisation and sentiment classification problems [2, 55, 85, 90, 102]. The problem that an SVM classifier tries to solve is described as “quadratic programming optimisation problem” [103]. This is explained by the following.

Let $\{(\underline{x}_1, y_1), (\underline{x}_1, y_2), \ldots, (\underline{x}_m, y_m)\}$ denote the set of training data, whereby $\underline{x}_i = (x_1, x_2, x_3, \ldots, x_n)$ denotes a document i. Each element of $\underline{x}_i$ is a binary that indicates feature presence or absence. $y_i \in \{-1, 1\}$ denotes the category of each document. A support vector machine algorithm is solving the following quadratic problem:

$$\begin{aligned} &\min_{\mathbf{w},\mathbf{b}} \left\{ \frac{1}{2} w^2 + C \sum_{i=1}^{n} \epsilon_i \right\} \\ &\text{s.t.} \quad \forall i: \; y_i(\langle w, x_i \rangle + b) \geq 1 - \epsilon_i \quad \epsilon_i \geq 0, \end{aligned} \tag{2.7.1}$$

where ϵ_i are the slack variables for the non-separable case, $\langle w, x_i \rangle$ is the dot-product of w and x_i, and $C > 0$ is the soft margin which controls the differences between margin b and $\sum_{i=1}^{n} \epsilon_i$. In other words, C is a penalty for the misclassified data. This penalty rises with the distance to the margin. w is the slope of the hyperplane which separates the data [128]. Therefore, SVM is a quadratic programming optimisation problem as it searches for the optimal hyperplane that separates the data and achieves the maximum margin to the positive examples on one side, and to the negative ones on the other side [103].

The speciality of SVM comes from the ability to apply a linear separation on the high dimensional non-linear input data. This is achieved by using an appropriate kernel function or the so called kernel trick [105]. The kernel trick means that the data is embedded in a higher dimension space using a mapping function, which makes it possible to separate the data linearly. For illustration see Fig. 2.1.

Fig. 2.1 demonstrates how the kernel trick works. In the left hand side of the figure are several points in one dimensional space. They can be only separated by a non-linear hyperplane. The kernel function maps those point into a two dimensional

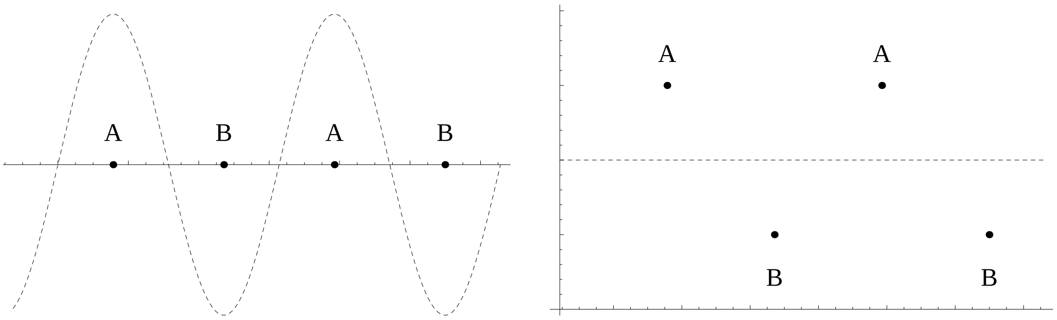


Figure 2.1: An overview of the mechanism of the kernel function. The data is mapped from a one dimensional to two dimensional spaces so that it can be separated linearly. The dash line represents the separator

space and that allows a linear separator to take a place.

The kernel function is expressed by $K(x_i, x_j)$, and it depends on the distance between x_i and x_j. The trick is that any N data points, can be separated linearly in a $N-1$ dimensional space [103]. Thus, after mapping the data into a higher dimensional space, we know that a linear separator must exist. SVM effectiveness is often affected by the types of kernel function that are chosen and tuned based on the characteristics of the data. The kernel function that is used in this thesis is radial basis function kernel (RBF) that is defined as

$$K(x_i, x_j) = \exp(-\gamma \|i - x_j\|^2), \gamma > 0. \tag{2.7.2}$$

The advantage of RBF is that the data can always be separated after it is mapped to a higher dimensional space [105]. It also requires only a few parameters (γ, C). Furthermore, this kernel can perform well in case of non-linearity between features and classes [46].

One could be faced with some issues while using SVM method. Each feature is a one dimension in the vector space. Features vectors in SVM are treated equally. This implicates that using SVM on a noisy or not processed texts is not very efficient, as many irrelevant features will be included. Another issue is the wide range of values that could create a domination of big values attributes over small values attributes especially if the data is not scaled [40]. Appropriate feature selection and scaling methods should be able to solve this problem and improve the SVM classifier prediction accuracy. In addition to those issues and as discussed earlier, in raw text there are some non-linearity between features and targeted classes. This problem could be solved by selecting an appropriate kernel function [46] or by including correlation concepts in the feature selection process.

In [2], SVM classifiers are used for sentiment analysis with several univariate and multivariate methods for feature selection, reaching 85-88% accuracies after using chi-squared for selecting the relevant attributes in the texts. A network-based feature selection method, that is, feature relation networks (FRN), helps to improve the performance of the classifier [2]. The pre-processing effect on the SVM classifier performance is explored in more details in Chapter 3.

2.7.1 Performance Evaluation

In text classification, the classifier evaluation is usually concerned with the classifier effectiveness rather than its efficiency [107]. That means it evaluates the correctness of the classifier predictability rather than its computational complexity. The common statistics that are used in text mining are precision and recall in addition to accuracy and F-measure [52, 99, 3, 39]. They are usually computed from a confusion table that presents the correctly and wrongly classified cases for all categories [107, 56]. The confusion table is presented in Table 2.1

	Class Negative	Class Positive
Predicted Negative	tn	fp
Predicted Positive	fn	tp

Table 2.1: The results of the classification presented in a confusion table

Table 2.1 presents the classification results where tn are true negative cases which is the number of correctly predicted negative documents. tp is true positive. That is the number of correctly predicted positive classes. fp is false positive which is negative classes predicted as positive, and fn is false negative, that is positive classes predicted as negative.

The evaluation metrics are computed based on the values in the confusion table as follows:

Accuracy is the number of correctly predicted documents out of all documents. It is computed by:

$$Accuracy = \frac{tp + tn}{tp + tn + fp + fn} \tag{2.7.3}$$

Precision is the number of true positive out of all positively assigned documents, and it is given by

$$Precision = \frac{tp}{tp + fp} \tag{2.7.4}$$

Recall is the number of true positive out of the actual positive documents, and it is given by

$$Recall = \frac{tp}{tp + fn} \tag{2.7.5}$$

Finally F-measure is a weighted method of precision and recall, and it is computed as

$$F - measure = \frac{2 * precision * recall}{precision + recall} \tag{2.7.6}$$

where its value ranges from 0 to 1 and indicates better results the closer it is to 1.

2.8 CONCLUSION

The effect of text pre-processing is well addressed in the information retrieval field. The pre-processing is divided into two processes in the literature: transformation and filtering. New experimental results are reported to address pre-processing in sentiment analysis and measure to which extent the usage of text pre-processing helps improving the classification results. The effect is measured separately for both processes. A combination of several existing transformation and filtering methods is suggested under a computational framework that achieves high prediction accuracies comparing to those achieved in previous studies.

The specific impact of the positive and negative words on the classification process especially on the SVM classifier has not been addressed before. This impact is investigated through the relationship between the existence of different polar words in a text and the SVM confidence in the prediction of that text.

As for financial news, the main problem of this field is the lack of labelled data. Moreover, all studies that investigates financial news sentiment tackle the problem from a news analytics perspective. They use ready sentimental scores, from leading companies in news analytics such as Reuters and Ravenpack, to investigate their impact on various economical variables. This thesis approaches this problem from a computational linguistics perspective and suggests two new models to predict financial news sentiment, in addition to a new method to label financial news.

Reader sentiment has not been approached before in the literature due to the difficulty of collecting the required data. In this thesis it is illustrated how financial news can be used to capture the reader sentiment. In this respect, the reader is assumed to be an active reader that takes actions to react to the news. In other words, the reader here is the investor. A new algorithm that utilises the reader sentiment is suggested to predict financial news sentiment.

Finally, the selection of domain dependent words is well utilised in cross domain analysis. Feature frequencies and mutual information are the most common methods to select domain dependent words. However, there are no clear way to identify the threshold on which those words are selected. In this thesis a new method is suggested to select the threshold that helps to separate domain dependent from domain independent words.

In conclusion, a machine learning approach will be used to classify the sentiment of product reviews and financial news using SVM in the analysis. The level of the analysis will be on a document level. Both views, the holder and the reader, are investigated in this thesis. Three key issues are addressed to improve the performance of sentiment classification, the impact of text pre-processing on sentiment analysis, and the use of text properties through financial news using events announcements and reader-based sentiment analysis, and finally cross domain sentiment analysis. The next chapter investigates the effect text pre-processing has on sentiment classification outcome.

CHAPTER 3

TEXT PRE-PROCESSING FOR SENTIMENT ANALYSIS

Although there have been diverse methods suggested for sentiment analysis for the various applications that are mentioned in Chapter 2 [29, 90, 118, 114], there are common characteristics that can be observed. The comprehension and accommodation of these characteristics can help understand the differences between different applications and corresponding methods and enable applications to be developed in a principled way. High dimensionality in texts is one of the text's main properties that make text pre-processing very important in text classification problems including sentiment analysis [107, 31]. Pre-processing transforms unstructured text into a machine processable input for text classifiers. Many researchers have emphasised the effect text pre-processing has on improving the classification accuracies [49, 23, 72]. The machine learning approach for text classification is a widely used approach [120, 107]. It is also the approach that is being taken in this chapter.

In this chapter, different text pre-processing methods are combined together to classify the sentiment of movies reviews. Experimental results are reported, which demonstrate that with appropriate feature selection and representation, sentiment analysis accuracies using support vector machines (SVM) may be significantly improved. The level of achieved accuracy is shown to be comparable to the ones achieved in topic categorisation although sentiment analysis is considered to be a much harder problem in the literature [90]. In addition, negative and positive features are shown to have different impact on the classification confidence with respect to the size of the corresponding documents. The impact of negative words is high on the classifier and that makes it easier to predict negative classes for documents with a lower number of features.

3.1 Framework

A computational framework for sentiment analysis that consists of three key stages is suggested. First, most relevant features will be extracted by employing extensive data transformation, and filtering. Second, the classifiers will be developed on each of the feature matrices that are constructed in the first step and the accuracies resulting from the prediction will be computed. Third the classifier's performance will be evaluated and compared to some previous results.

The most challenging part of the framework is feature selection that will be discussed here in some depth. Firstly, transformation is applied on the data. That includes HTML-tag clean up, abbreviation expansion, stopwords removal, negation handling, and stemming. Natural language processing techniques are used to perform transformation on the data. As a result of this stage, three different feature matrices are computed based on different feature weighting methods: feature frequency (FF), feature presence (FP), and term frequency-inverse document frequency (TF-IDF). Secondly, a filtering process is applied, in which the χ^2 statistics is computed for each feature within each document, and then a threshold of χ^2 is chosen to select the most relevant features based on a critical p-value level. At the end of this stage, another three feature matrices are constructed based on the same previous weighting methods.

$$\begin{tikzcd} F^{(b)} \arrow[r] & F^{(t)} \arrow[r] & F^{(f)} \end{tikzcd}$$

Figure 3.1: The diagram illustrates the order of the three instances of the data matrices $F^{(i)}, i \in \{b, t, f\}$ that contain the document-feature relationship before transformation, after transformation, and after filtering respectively. Starting with the most raw matrix on the left.

The relationship between features and the containing documents is denoted with adjacency matrices

$$F^{(k)} = \left(F^{(k)}_{ij}\right)_{\substack{i=1,\dots,n\ , \\ j=1,\dots,m}} \quad k \in \{b, f, t\}. \tag{3.1.1}$$

Each entry (i, j) is the weighted value of feature i in document j. The details of the different feature matrices $F^{(b)}, F^{(t)}$, and $F^{(f)}$ are described in the following. $F^{(b)}$ is the most bare matrix. It contains the feature-document relationships before any pre-processing is carried out. The next more explanatory matrix is $F^{(t)}$ that is constructed after transformation. Finally $F^{(f)}$ is the matrix that is composed after filtering. Fig. 3.1 summarises the relationship between the different matrices.

As explained earlier, for each stage the matrices are constructed over three different weighting methods are used FP, FF, and TF-IDF. For FP, the (i, j)-th entry of the matrix is 1 if the feature i is present in the document j and is 0 otherwise. For FF the (i, j)-th entry of the matrix denotes the frequency of feature i in document j. Finally for TF-IDF, the (i, j)-th entry of the matrix is computed as in Eq. (2.6.1) for feature i in document j.

The data consists of two data sets of movie reviews. One was first used in [90] containing 1400 documents (700 positive and 700 negative). This set will be referred to as Dat-1400 in the following. The other data set was constructed in [88, 2] with 2000 documents (1000 positive, 1000 negative). It will be referred to as Dat-2000 in the course of this thesis. Both sets are publicly available[1]. Although the first

[1] http://www.cs.cornell.edu/people/pabo/movie-review-data/

set is included in the second set, they are treated separately. The reason for this is that each data set has different feature sets sizes, and they influence the polarity of the text differently. This will be demonstrated with some extensive experiments in Subsection 3.2.3. Furthermore this separation allows a fair comparison with the results from different studies that used these data sets separately. Only unigrams are considered in this thesis. The data is processed as follows.

3.1.1 DATA TRANSFORMATION

The data does not contain any HTML tags. Further processing uses the techniques of natural language processing. The abbreviations are expanded using pattern recognition and regular expression techniques. As for stopwords, a stoplist is constructed from several publicly available standard stoplists [124], with some changes related to the specific characteristics of the data. For example, the words *film, movie, actor, actress, scene* are non-informative in movie reviews data. They are considered as stop words because of their frequent appearance in the text on one hand, and because they are used specifically in the movie domain which makes them domain specific stopwords on the other hand. For stemming, the Java class Stemmer offered by the library Spider is implemented.

As for handling negation, at first, it is handled as it is suggested in [90]: all negation words are tagged with the following words till the first punctuation mark occurrence. Then this tag is used as a one feature in the classifier. By comparing the results before and after adding the tagged negation to the classifier there was not a difference in the classifier performance. This conclusion approves the findings of [32]. The reason is that this way is producing more noise as each tag will be considered as a feature. With this low frequency, those tags will not be included in the final feature sets. For that reason, the number of tagged words after the negation is set to three then to two words after the negation, and this allows for more negation phrases to be included in the final features sets.

It is worth to mention that negation of negated words is treated in the same way. For instance in a sentence like " the movie was not unlikable", the extracted negated tag will be "NOT_unlikable", and this tag is then treated as any other feature in the text. This tag needs to occur in certain frequencies over several documents to be considered as a feature of impact on the text orientation. Another issue is that some sentences may contain negative words but express a positive sentiment or vice versa. This issue could be a limitation of a sentiment analysis classifier that does not take a linguistic approach in consideration. However, in the case of this thesis, this limitation can be ignored due to the fact the classification is applied on a document level while they cannot be ignored in the case of sentence level classification, and therefore, tricky sentences can be eliminated throughout the feature selection process or the pattern recognition in the classification process. This does not imply that including those cases will not improve the classification accuracies. A linguistic approach can be combined with the SVM classifier to investigate the effect of such sentences on the classifier's prediction.

In addition, a stemmer is applied on the documents to reduce redundancy. This reduces the number of features from 10450 to 7614 in Dat-1400, and from 12860 to 9064 features in Dat-2000.

At the end of the transformation stage, three feature matrices $F^{(t)}$ are constructed for each of the datasets based on three different types of features weighting: TF-IDF, FF, and FP as previously described. A number of experiments are carried out on the $F^{(t)}$ feature matrices of Dat-1400 and Dat-2000. The results are shown in Section 3.2

3.1.2 FILTERING

The method that is used for filtering is the univariate χ^2 method. As previously explained, χ^2 is a statistical analysis method used in text categorisation to measure the dependency between the word and the category of the document it is mentioned in. If the word is frequent in many categories, chi-squared value is low, while if the word is frequent in few categories then chi-squared value is high.

The value of χ^2 statistic is computed for each feature that is included in the feature matrix $F^{(t)}$. After that, based on a 95% confidence level of the χ^2 statistic, a final set of features is selected in both datasets, resulting in 776 out of 7614 features in Dat-1400, and 1741 out of 9064 features in Dat-2000. The two sets are used to construct the $F^{(f)}$ matrices on which the classification was conducted. At this stage, each data set has three $F^{(f)}$ feature matrices resulting from the three different weighting methods: FP, FF, and TF-IDF.

3.1.3 Classification process

To investigate the effect of text pre-processing on the classifier's performance, the classification process is applied on all previously constructed matrices: before pre-processing $F^{(b)}$, after transformation $F^{(t)}$, and after filtering $F^{(f)}$. The SVM classifier is used with a Gaussian radial basis kernel function (RBF) which has the parameter γ that controls for the area in which the support vector has an effect in the data space. Due to the sensitivity of the SVM classifier to the kernel parameters (C, γ), the classifier needs to be tuned, that is, to choose the optimal parameter set for each data set on each level. The optimal parameters are the combination of (C, γ) that achieves the best possible performance. The classifier is tuned using a grid search over given ranges of the two parameters as follows. Define a set of pairs of different values of (C, γ). For each pair, do the train the classifier and compute the classification error. In the end choose the set of values that achieves the minimal error estimate and set those values as the optimal parameters.

For the classification process, each set is divided into two parts: one for training and the other for testing, by ratio $9 : 1$, that is $9/10$ parts were used for training and

$1/10$ for testing. Afterwards, training is performed with 10 folds cross validation for classification. SVM is applied by using the machine learning package ‘e1071’ in R [34]. Results are shown in Section 3.2

3.2 Experiments and Results

In this section the results of several experiments are reported to assess the performance of the classifier. The classifier is implemented on each of the feature matrices resulting from data transformation and filtering. Three different impacts on the classifier performance are distinguished: transformation, filtering, and feature set size.

3.2.1 Impact of Transformation

Ref. [87] argues that “standard machine learning classification techniques, such as support vector machines (SVMs), can be applied to the entire documents themselves” and this is why the classifier is applied on the entire texts with no pre-processing or feature selection methods in [90, 87]. Therefore, to allow for a fair comparison with previous results, based on the tuned kernel parameters that are used in this stage ($\gamma = 0.001$ and $C = 10$), the first input of the classifier is the bare matrix with no pre-processing $F^{(b)}$. Then, to assess the impact of text transformation on the classification performance, the classifier is applied on the data after transformation $F^{(t)}$. The experiments are carried out on all feature weighting types. The results and comparison to [90] are shown in Table 3.1.

Table 3.1 compares the classifier performances resulting from the classification on both, not pre-processed, and pre-processed data for each of the features matrices (TF-IDF, FF, FP). Furthermore it compares these results to those that are achieved in [90]. The comparison is based on the accomplished accuracies and the metrics calculated in Eqs. (2.7.4), (5.4.1), (2.7.6) .

	TF-IDF		*FF*			*FP*		
	$F^{(b)}$	$F^{(t)}$	$F_1^{(b)}$	$F_2^{(b)}$	$F^{(t)}$	$F_1^{(b)}$	$F_2^{(b)}$	$F^{(t)}$
Accuracy	78.33	81.5	72.7	76.33	83	82.7	82.33	83
Precision	76.66	83	NA	77.33	80	NA	80	82
Recall	79.31	80.58	NA	76.31	85.86	NA	83.9	83.67
F-Measure	77.96	81.77	NA	76.82	82.83	NA	81.9	82.82

Table 3.1: The classification accuracies in percentages on Dat-1400. The column $F_1^{(b)}$ refers to the results reported in [90], and $F_2^{(b)}$ refers to the results in this thesis with no pre-processing, and $F^{(t)}$ refers to the results after transformation, with optimal parameters $\gamma = 10^{-3}$, and $C = 10$

Table 3.1 shows that accuracies vary across the different weighting methods. For the data without any pre-processing $F^{(b)}$, using FP gives the best results among the other methods. This agrees with the results in [90]. However, data transformation does not have an impact on the classifier performance using FP: the accuracy changes only slightly from 82.33% in $F^{(b)}$ to 83% in $F^{(t)}$. On the other hand the changes in accuracies using FF and TF-IDF from $F^{(b)}$ to $F^{(t)}$ is more pronounced. For FF, the accuracy rises from 76.33% in $F^{(b)}$ to 83% in $F^{(t)}$, while it changes from 78.33% in $F^{(b)}$ to 81.5% in $F^{(t)}$ using TF-IDF.

Table 3.1 shows that although the accuracy accomplished in the FP matrix is close to the one achieved before and in [90], there is a big improvement in the classifier performance on the TF-IDF and FF matrices after transformation is applied. This shows the importance of stemming and removing stopwords in achieving higher accuracies in sentiment classification. It is fair to mention that in order to be able to use the SVM classifier on the entire document without any pre-processing, one should design and use a kernel for that particular problem [106].

3.2.2 Impact of Filtering

After the transformation, the three different matrices that are constructed after the filtering $F^{(f)}$ are used for classification. The kernel parameters that are used in this stage ($\gamma = 10^{-5}$ and $C = 10$). The results can be found in Table 3.2. Those results are high comparing to what is achieved in previous experiment and in [90]. Feature selection based on their χ^2 statistic value leads to dimensionality and noise reduction in the text which has a big impact on the classifier performance.

Table 3.2 presents the accuracies and evaluation metrics of the classifier performance before and after χ^2 is applied.

	TF-IDF		*FF*		*FP*	
	$F^{(t)}$	$F^{(f)}$	$F^{(t)}$	$F^{(f)}$	$F^{(t)}$	$F^{(f)}$
Accuracy	81.5	92.3	83	90	83	93
Precision	83	93.3	80	92	82	94
Recall	80.58	91.5	85.86	88.5	83.67	92.16
F-Measure	81.77	92.4	82.83	90.2	82.82	93.06

Table 3.2: The classification accuracies in percentages before and after using chi-squared on Dat-1400, with optimal parameters $\gamma = 10^{-5}$, and $C = 10$

Table 3.2 shows a significant quality increase in the quality of the classification, with the highest accuracy of 93% achieved when the FP matrix is used, followed by 92.3% using TF-IDF and 90.% using FF matrices. Likewise, the F-measure results are very close to 1, which indicates a high classification performance. To the best of my knowledge, comparable results are not reported in document level sentiment analysis using chi-squared in previous studies.

To conclude, the use of transformation and then filtering on the texts data reduces the noise in the texts and improves the performance of the classification. Fig. 3.2 shows how the prediction accuracies of SVM gets higher when the data changes from not pre-processed to pre-processed.

Fig. 3.2 compares the classifier performances resulting from the classification on each of $F^{(b)}$, $F^{(t)}$ and $F^{(f)}$ matrices. The comparison is based on the accuracies and the metrics calculated in Eqs. (2.7.4)-(2.7.6). The exact values can be found in Table. 3.3. All three measures of goodness of classification show roughly the same pattern. There is a slight increase of goodness when $F^{(t)}$ is used instead of $F^{(b)}$. However, using $F^{(f)}$ leads to a big jump of goodness which shows that filtering is

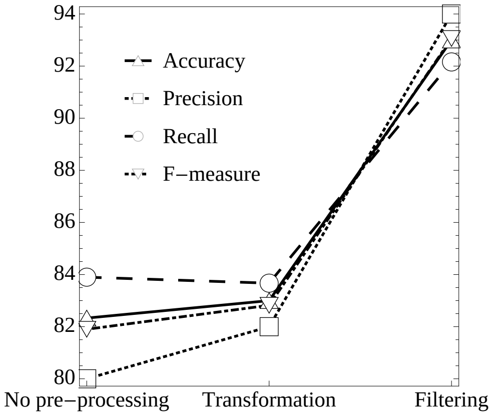


Figure 3.2: Goodness of classification measured in terms of accuracy, precision, recall and F-measure for the three different data matrices $F^{(b)}$, $F^{(t)}$ and, $F^{(f)}$. It is clear that in particular the filtering leads to a significant increase in the classification quality.

essential for robust outcomes.

	No pre-processing	Transformation	Filtering
Accuracy	82.33	83	93
Precision	80	82	94
Recall	83.9	83.67	92.16
F-Measure	81.9	82.82	93.06

Table 3.3: The classification best accuracies in percentages on Dat-1400 with FP weighting method. The column no pre-processing refers to the classification results without any pre-processing, and transformation column refers to the classification results after data transformation, while the last column refers to the classification results after filtering the data using χ^2 with a 95% level of confidence.

As for Dat-2000, the data is pre-processed until the feature matrix $F^{(f)}$ is constructed using the three weighting methods. Those matrices are the input of SVM classifier that delivers a high accuracy of 93.5% using TF-IDF followed by 93% in FP and 90.5% in FF (see Table 3.4).

	TF-IDF	*FF*	*FP*
Accuracy	93.5	90.5	93
Precision	94	89.5	91
Recall	93.06	91.3	94.79
F-Measure	93.53	90.4	92.87

Table 3.4: Best accuracies in percentages resulted from using chi-squared on 2000 documents, with optimal parameters $\gamma = 10^{-6}$, and $C = 10$

Using the same data set, a feature relation networks selection based method (FRN) is proposed in [2] to select relative features from Dat-2000 and improve the sentiment prediction using SVM. The achieved accuracy using FRN 89.65% , compared to an accuracy of 85.5% when the chi-squared method as a feature selection method. The features that are used in [2] are of different types including different N-grams categories such as words, POS tags, and others, while only unigrams are used in the present work. This demonstrates that using unigrams in the classification has a better effect on the classification results compared to other feature types. This is consistent with the findings of [90].

3.2.3 THE IMPACT OF FEATURE SETS SIZES

DAT-1400

In a different set of experiments, the effect of the features set size on the classifiers performance is explored. Here, only FP as a weighting method is used due to its high performance in the previous set of experiments. Furthermore, only the final feature set in matrix $F^{(f)}$ is employed in this set of experiments. Defining four different critical p-values of the χ^2 statistic (0.05, 0.01, 0.005, 0.001) leads to four features sets of sizes (776, 449, 325, 174) respectively (see Fig. 3.4).

The $F^{(f)}$ matrix is then constructed for each of those feature sets. For each of the $F^{(f)}$ matrices the classifier is run 60 times, whereby the training set is selected randomly from the Dat-1400 each time. The outcome of the classification process is averaged over those 60 iterations. Results are presented in Table. 3.5.

The number of iterations for these experiments has been chosen heuristically by weighing the computational complexity against the added information that is contained in additional repetitions. Fig. 3.3 illustrates the relationship between the number of iterations N and the convergence of the cumulative average of accuracies $\mu_N = 1/N \sum_{j=1}^{N} \mathrm{ACC}_j$.

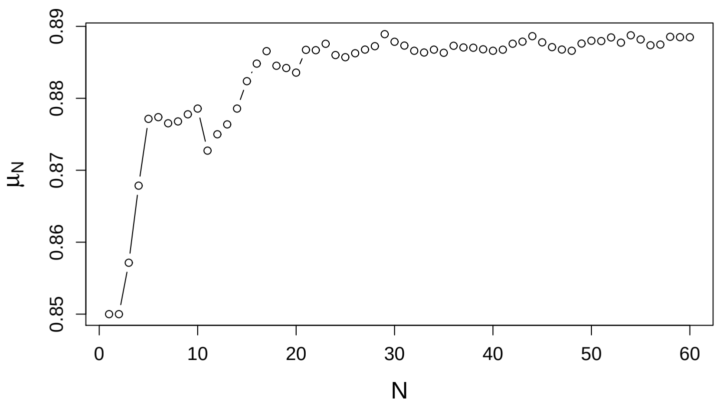


Figure 3.3: The relationship between the number of iterations N and the convergence of the cumulative average of accuracies μ_N after each iteration .

p-value	features	*Accuracy*	*Precision*	*Recall*	*F-Measure*
0.05	776	88.79	90.43	87.71	88.98
0.01	449	83.83	83.59	84.16	83.80
0.005	325	83.083	82.30	83.81	82.95
0.001	174	81.667	80.95	82.32	81.51

Table 3.5: The classification average accuracies in percentages after using χ^2 with different critical p-values and different numbers of features. Dat-1400

Table. 3.5 shows the effect of the number of features on the classification performance. It illustrates that the performance is increasing for the higher p-values. This means that despite the previously noticed increase in the classification performance with decreasing number of features, there is a limit to this improvement. That is because the number of features that satisfy these more restrictive conditions is decreasing to an extent that it does not contain enough information for the classifier to classify the documents. Fig. 3.4 illustrate this relationship.

Fig. 3.4 depicts the relationship between the frequency of a feature i over all documents $f_i^{(t)}$ and the corresponding χ^2 values that measure dependency of the feature to a certain category. To be more precise

$$f_i^{(t)} = \sum_{i=1}^{m} F_{ij}^{(t)}. \tag{3.2.1}$$

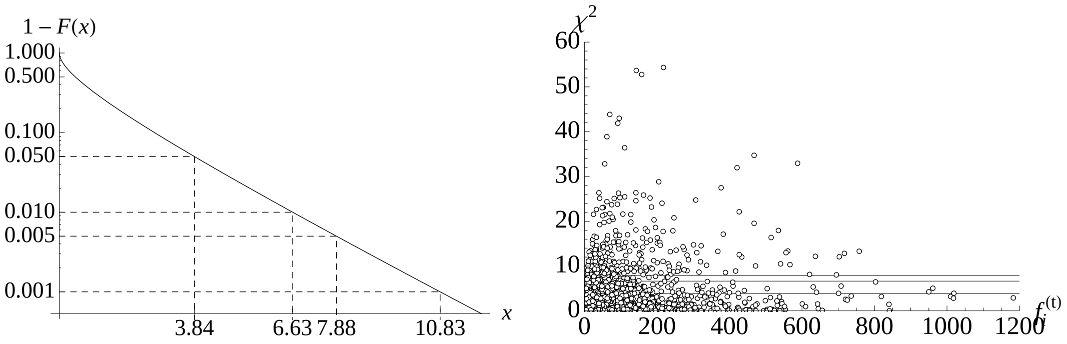


Figure 3.4: The scatter-plot illustrates the relationship between the frequency of the feature f_i and the features' significance in the classification procedure, measured by the χ^2 test statistic. The three vertical lines correspond to different p-values, starting from the lowest line $p = 0.05(\chi^2 = 3.84)$, $p = 0.01(\chi^2 = 6.63)$, $p = 0.005(\chi^2 = 7.88)$, $p = 0.001(\chi^2 = 10.83)$. Most desirable are features in the upper-right corner. Those are highly significant and frequently found.

It is clear from Fig. 3.4 that the number of features with very high χ^2 (low p-value) is low. However, for the standard significance levels that are marked with horizontal lines, there is still a large number of features that satisfy these constraints. Another observation is that there are no features that are highly significant and very frequent at the same time. This finding underlines the trade-off that one faces in these classification problems.

Investigating further into the classifier performance, the relationship between the number of features in each document and the distance to the separating hyperplane Δ_d is explored. The predicted documents are split into two parts. The first part contains all the negative documents: the true negative (tn) and false positive (fp) predicted documents. The second one contains all the positive documents: the true positive (tp) and false negative (fn) predicted documents. The distances to the separating hyperplane are computed for each document. For the following, this distance is understood as the confidence of the prediction. The following convention is used: true negative documents and false negative are identified by negative distance, while true positive and false positive are identified by positive distances. Results are shown in Fig. 3.5.

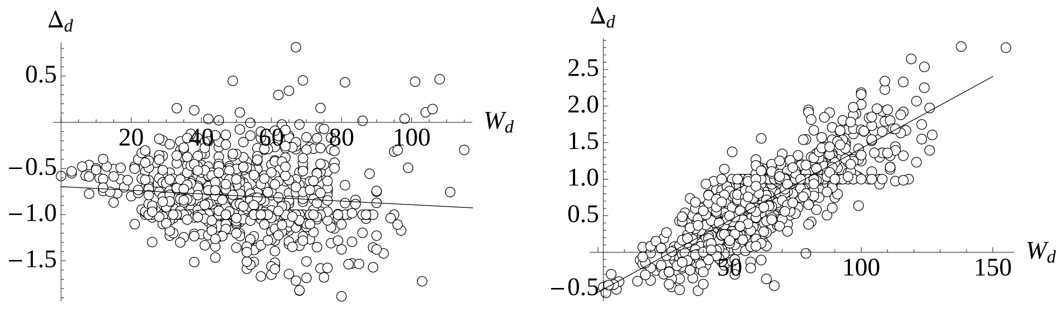


Figure 3.5: Scatter plots depicting the relationship between the number of features in each document W_d and the distance to the separating hyperplane Δ_d for each document in the negative category (left-hand side) with regression line $\Delta_d = -0.697 - 0.0019W_d$ ($R^2 = 0.008$). For the positive class (right-hand side) with regression line $\Delta_d = -0.548 + 0.019W_d$ ($R^2 = 0.691$)

Fig. 3.5 shows that for the negative category, only a very weak negative correspondence between the number of features in each document and the distance of this document to the separating hyperplane can be found. This implies that a higher number of features in a document does not lead to a better prediction in this category. This finding is very different from the positive category, where there is a better linear correlation between the number of features in a document and the distance to the separating hyperplane. The figure illustrates in the right-hand side that a higher number of features corresponds to a larger distance from the separating hyperplane in the negative category. These results indicate that the prediction of the negative category is relatively easier because even for a low number of features the prediction gives correct results. In other words, fewer negative words are needed to predict the text as negative with high confidence, while more positive words are needed to predict the positive class with high confidence. This is confirmed by the finding that most significant words for this data set have a negative sentiment (see Table. 3.6).

A more detailed look at Fig. 3.5 on the left-hand side reveals that the distance for true negative documents ranges between zero and 2. On the other hand, the distance for true positive in Fig. 3.5 on the right-hand side ranges between 0 and 3. This means that the negative documents are more similar to each other than the positive documents are.

boring	-0.25	suppose	0.05	different	0.22
stupid	-0.43	world	-0.38	hilarious	-0.12
worst	-0.50	detail	0.01	bland	-0.14
lame	-0.23	outstanding	0.37	intense	0.00
dull	0.02	terrible	-0.66	subtle	0.05
mess	-0.10	touch	-0.002	fail	-0.16
laughable	0.43	ridiculous	-0.12	unfunny	-0.25
solid	0.24	view	-0.05	painfully	-0.08
great	0.11	uninteresting	-0.14	lifeless	-0.23
wonderfully	0.38	memory	0.01	poorly	-0.625

Table 3.6: First 30 most significant words starting from left up corner down, with their sentiment scores according to Ref. [8]

In Fig. 3.6, the boxplots illustrate the average the goodness-of-classification measures, together with the dispersion of the results over different iterations for different p-values from Table 3.5. Curiously, the dispersions in all of the box plots are larger than anticipated. The reason for such observation is related to several issues. First, the training set is selected randomly at each of the 60 iterations. Second, the documents have different length and different feature distributions. Since the classification performance is affected by the size of the document and the feature distribution as illustrated earlier, the choice of the training set reflects on the classification performance and is shown in this large dispersion.

DAT-2000

To have a closer look at a larger data set, the same experiments are run on Dat-2000. With the same four critical values of χ^2 statistic (0.05, 0.01, 0.005, 0.001) four different feature sizes are generated (1741, 799, 603, 338). The results of 60 iterations of the classifier on the $F^{(f)}$ matrix with FP weighting methods are presented in Table 3.7.

Looking at Table 3.7, the changes in the classification performance with the change of the number of features are noticed. The highest achieved accuracy of 89.59%

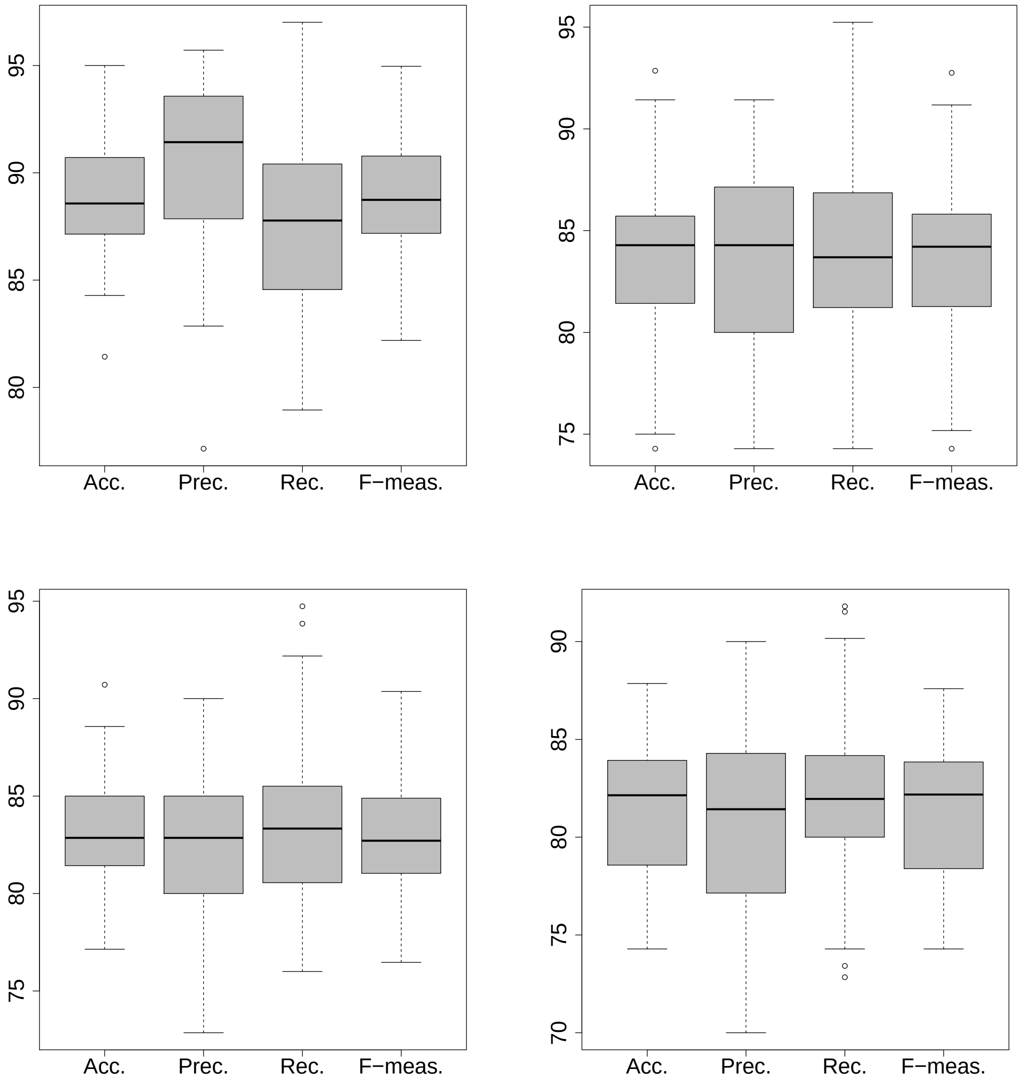


Figure 3.6: boxplots of the goodness-of-classification measures for different p-values from top left 0.05, 0.01, 0.005, 0.001 respectively

is coupled with the highest p-value at a 95% level of significance. As the p-value decreases to 0.01 the accuracy drops to 85.62%, then increases slightly to 86.2% and drops back to 85.06% at p-values 0.005 and 0.001 respectively. To investigate the significance of this slight increase, the non-parametric Wilcoxon signed rank test [130] is implemented. The Wilcoxon singed rank test measures for two samples whether their means are significantly different from each other. The results show that the slight increase in the accuracy is not statistically significant, and in this

p-value	features	*Accuracy*	*Precision*	*Recall*	*F-Measure*
0.05	1741	89.59	89.38	89.79	89.56
0.01	799	85.62	84.93	86.21	85.49
0.005	603	86.2	85.61	86.72	86.1
0.001	338	85.06	84.46	85.56	84.96

Table 3.7: The classification average accuracies in percentages after using χ^2 with different critical p-values and different numbers of features. Dat-2000

context, it is not related to the decrease in the number of features. Therefore, the conclusion to be drawn is that after a certain decrease in the number of features the quality of the classification drops because there is not enough information to be fed to the classifier.

Looking into the relationship between the number of features in each document and the distance to the separating hyperplane (see Fig.3.7), the same pattern as in Dat-1400 can be noticed. The figure reveals that there is a linear correlation between the number of features in true positive documents and the distance to the separating hyperplane. Hence, the more features there are in the positive documents the more information is available to make the classification performance better. The opposite conclusion can be drawn for negative documents as the number of the features in true negative documents does not correlate with the distance to the hyperplane. Documents with low features are predicted correctly and that reflects the strong impact negative words have on the classification prediction of negative documents.

Another observation from Fig. 3.7 is the difference in the distance range between the positive and negative categories. The distance range in the true positive documents is between 0 and 6, while it is between 0 and 2 in the true negative documents. Again, this reveals that also in this data set the negative documents are more similar to each other than the positive document. It also means that for positive documents, the confidence in prediction rises with the increase of document length.

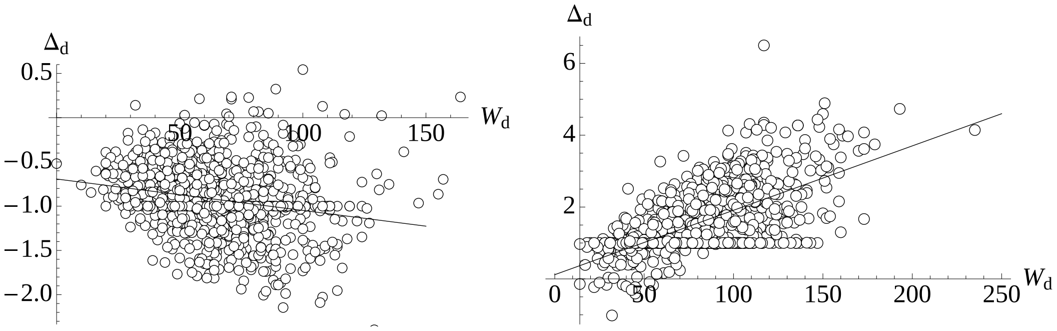


Figure 3.7: Scatter plots depicting the relationship between the number of features in each document W_d and the distance to the separating hyperplane Δ_d for each document in the negative category (left hand side) with regression line $\Delta_d = -0.69 - 0.0.0036W_d$ ($R^2 = 0.04$). For the positive class (right hand side) with regression line $\Delta_d = -0.18 + 0.018W_d$ ($R^2 = 0.397$)

3.3 CONCLUSION

In this chapter, the sentiment of online movie reviews is investigated. A combination of different pre-processing methods are employed to reduce the noise in the text in addition to using the χ^2 method to remove irrelevant features that do not affect the text's polarity. Extensive experimental results have been reported, showing that, appropriate text pre-processing methods including data transformation and filtering can significantly enhance the classifier's performance. The level of accuracy that is achieved on the two data sets is comparable to the sort of accuracy that can be achieved in topic categorisation, a much easier problem.

Furthermore, the relationship between the number of features in a text and the prediction outcome of the classifier is investigated. It turned out that in the positive category the longer the document is, the higher are the probability and the confidence that the document is classified correctly. As for the negative category, the probability of correct prediction is high even for small sized document. Furthermore, the impact of negative words is high on the classifier and that it is easier to predict negative classes for a lower number of features.

Financial websites and blogs are a source of a different type of text data in the huge pool of social media. They have some properties that allow to extract some information that can be useful to improve sentiment classification. How announced events in the news can predict the news sentiment, how investor’s sentiment is related to stock price fluctuations, and how the investor’s opinion can be translated into a signal for buying or selling are the topics that will be addressed in the next chapter.

CHAPTER 4

TEXTUAL PROPERTIES: SENTIMENT ANALYSIS IN FINANCIAL NEWS

4.1 INTRODUCTION

An important part of the information that can be extracted from the news is the sentiment score [79]. As a branch of the so called News Analytics, sentiment analysis for news is widely researched in academia and industry. Whenever a piece of news about a publicly listed company is issued, it contains information about this company. This information can be good, bad, or neutral, which is usually reflected in the stock returns. Therefore, studying that information is important to identify any effects they might have on the stock market returns. Next to this importance, financial news sentiment analysis allows to capture the reader sentiment. The reader in this respect is the investor in the stock market. The investor receives the news sentiments and transfers it into a buy, sell, or hold decision, which in turn affects the stock returns.

Referring to the news effect brings to one's mind the question of market efficiency.

An efficient market is a market where prices of stocks “fully reflect” the available information [101]. The market efficiency hypothesis (MEH) has been widely tested and proved to be true with very few exceptions [50]. Market efficiency is of three types, weak, semi-strong, and strong forms. The market is considered to be weakly efficient if it fully reflects the information of the historical stock prices. The market is said to be semi-strongly efficient when it reflects in addition to the past information, the information released publicly. While it is classified as strong efficient market if it incorporate historical, public and private information [101]. Under the MEH, once the information is known to everybody in the market, the bids on the related stocks are made. Any similar information that arrives later on to the market should not affect the stock price. This hypothesis is not always true.

In the 27th of November, 2014, the New York Times (NYT) published a story about the Medallion Financial company with a negative sentiment, telling that the medallion prices are falling[1]. This piece of information was not new. It was known to the investors that the prices were falling and that the taxi market had some weaknesses. This was published in official statistics, accessible by any investor that was interested. According to the MEH, the NYT article should not have had an affect the stock price of Medallion Financial because the information was publicly available before the article was published. On the contrary, the stock price of Medallion Financial closed with more than 7% decrease. One possible reason for this market inefficiency, is that Medallion Financial is a small company and research efforts are more attracted by bigger companies[2]. Still, this raises the question about the validity of the market efficiency hypothesis in general. What could be addressed here is that the market is driven by the information that is published in various sources including financial news, the main interest of this chapter.

[1] http://www.nytimes.com/2014/11/28/upshot/under-pressure-from-uber-taxi-medallion-prices-are-plummeting.html?_r=0&abt=0002&abg=1

[2] http://www.nytimes.com/2014/12/04/upshot/how-our-taxi-article-happened-to-undercut-the-efficient-market-hypothesis.html?action=click &pgtype=Homepage&module=c-column-middle-span-region&region=c-column-middle-span-region&WT.nav=c-column-middle-span-region&_r=1&abt=0002&abg=0

One piece of the information that can be extracted from the news is the news event, the main content in the news, for example, bankruptcy, merger, acquisition, and so on. In event studies, the effect of an event's occurrence on the stock returns is measured. This concept was introduced by Fama et al. in 1969 to provide a proof of the stock returns' response to the information (see [37]). The connection to the present thesis is that it focuses on finding an impact on the stock's return emerging from the information mentioned in the news everyday. It is assumed that news has an effect within one trading day. This implies the assumption that the effect of the events is short-lived, since it is measured on short returns window in comparison to longer windows like two days or a week.

In this chapter, two different approaches are presented to extract the news sentiment based on two properties of financial news. First, based on events occurrences in the news, a new method that uses news events in sentiment classification is proposed. Second, based on the ability to capture the reader sentiment in financial news, a new method is proposed to label news items automatically, that is the Reader Sentiment Indicator (RSInd). The labelled news is then pre-processed and used as an input for SVM classifier to classify their sentiment. The first approach will be referred to as event-based sentiment, and the second approach as reader-based sentiment in the course of this thesis.

This chapter is structured as follows. First, two different frameworks to approach the problem are introduced in Section 4.2. Data sets and data collection are explained in Section 4.3. Then, the event-based sentiment analysis model, experiments and results are reported in Section 4.4. In Section 4.5, the reader-based sentiment model is suggested and SVM is used to predict news sentiment. The model is then validated as a trading strategy.

4.2 APPROACHING THE PROBLEM

In the following, two different novel methods are presented to approach the financial news sentiment classification problem. The first approach is event-based. Events are extracted from each news item in the data set about publicly listed companies. Then, the correlation between event occurrences and stock market returns is investigated. Once this correlation is established, an event-based sentiment model can be constructed. This correlation has been addressed in a different context economic research, but has not been utilised before to generate sentiment labels for financial news.

The second approach is reader based. It takes the news reader sentiment (investor) into consideration, and uses it to label the news into two categories: positive and negative. Then, the framework that is introduced in Chapter 3 is used to predict the news sentiment. This approach is validated against random trading strategies. Reader sentiment has not been investigated before in the literature due to the difficulties in obtaining the corresponding data. In this chapter, this issue is approached for the first time using the properties of financial news.

4.3 DATA SETS

The whole data set consists of three collections of records. The first one is a list of companies with their ticker symbols, index association and country of origin. This data is collected from freely available sources from the internet. The data consists of over 4500 stocks from various indices and stock exchanges from the USA, Germany, and the UK.

A second collection of records stores the stock market prices for every company of the first data set and all associated indices on a daily basis. This set contains the open, high, low, close, and volume data.

The third data set consists of more than 120,000 of news items. Every news item is a single record on a table, which contains information about publication date, publisher, the whole HTML page which contains the news article, as well as information about the news source, the date of crawling, and information from further processing of the raw data, which will be exemplified later. An excerpt of this table can found in Table A.1 in Appendix A.

4.3.1 Data Collection

In the present chapter, news articles from the period between 01/06/2011 and 09/09/2011 are analysed. News articles were collected from various sources which were obtained via a Yahoo Interface (YQL) on a daily basis, and then stored for further evaluation.

The type of news that is used in this thesis is formal news. That covers any topic proposed by a reliable news publisher, that is announced on TV, radio, or in newspapers and includes online versions [79]. News texts are provided by different services/API's of Yahoo!. Initially, this analysis was conducted only for UK based companies, listed in the FTSE100. During the first period of research, the collected data showed information for other different companies from various indices all over the world. The crawler collects next to the main article, other articles that are related to the corresponding company, for instance: articles about competitors in domestic or international markets, or similar companies in other countries, or companies that employ the same marketing strategies, etc. This is not surprising since the publicly listed companies are operating internationally, and are therefore affected by events caused by companies which are not traded in the FTSE100. Therefore, the analysis does not take specific indices in consideration. It depends solely on the available companies that are mentioned in the data.

To retrieve the text from the internet, a web-crawler is written to read the news automatically from the web everyday and save it to the database. This program was exposed to some technical problems regarding the availability of the webpage. These problems were solved by saving the content as HTML text to provide an ability to come back to the page for any sudden requirement or further processing. The downloaded amount of news was limited by Yahoo! API terms and conditions.

4.3.2 Refining Data

The data is cleaned first from HTML-tags. Afterwards, the news articles are extracted along with the following additional information: date of crawling, date of publishing, title and the corresponding company. That information is stored in a news items database. (see Table A.1). Afterwards, the data is refined differently for the two previously introduced approaches. For event based sentiment, events and related information are extracted using NLP and OpenCalais. More details can be found in Section 4.4. In a different set of experiments that address the reader-based sentiment, the news items are labelled using RSInd. Afterwards, all items are preprocessed under the framework that is suggested in Chapter 3. More details can be found in Section 4.5.

4.4 Event Based Sentiment Analysis

In this section the relationship between events that are mentioned in the news, and the stock market prices fluctuations is investigated. To achieve that, events need to be extracted first from the news as follows.

4.4.1 Event Extraction

After a news article is obtained from the news items database, it goes through two processes. In the first one, OpenCalais, a free online product provided by Thomson

Anniversary	RadioProgram	CompanyInvestment	Indictment
City,CompanyLocation	RadioStation	CompanyLaborIssues	IPO
Company	Region	CompanyLayoffs	JointVenture
Continent	SportsEvent	CompanyLegalIssues	ManMadeDisaster
Country	SportsGame	CompanyListingChange	Merger
Currency	SportsLeague	CompanyMeeting	MovieRelease
EmailAddress	Technology	CompanyNameChange	MusicAlbumRelease
EntertainmentAwardEvent	TVShow	CompanyProduct	NaturalDisaster
Facility	TVStation	CompanyReorganization	PatentFiling
FaxNumber	URL	CompanyRestatement	PatentIssuance

Table 4.1: OpenCalais Possible Entities

Reuters[3], is applied. OpenCalais is a tool that is used to extract semantic metadata from texts. It has an API which is accessible from JAVA. OpenCalais explores the given text, and returns several kinds of information. These are: Entity, Fact, Event Index and Definitions, Generic Relation Extraction, Entity Relevance Score, Social Tags, Document Categorisation, and Entity Disambiguation[4]. Entities, facts and events are most interesting for the analysis in this thesis, as they are focused on entities such as companies which are associated with the events. OpenCalais is able to recognise many entities, those can be found in Table 4.1. The entity that is mostly used in this chapter is “company”.

As for events, OpenCalais recognises the events listed in Table 4.2, which contain many events that might be relevant to market outcomes[5]. However, for this study, the set of events is adapted to the available data. For low frequency events, it is not possible to obtain a sufficiently sized sample of news. These events are therefore omitted in the further analysis.

The OpenCalais API returns output in four forms: micro-formats, simple format, JSON, and N3. Each of them has different characteristics. For the present thesis JSON[6] has been chosen because it is easier to be handled, read and parsed than the other formats. An example of the outcome is given in Listing B.1, in Appendix B.

[3] http://www.opencalais.com

[4] http://www.opencalais.com/documentation/calais-web-service-api/api-metadata

[5] See http://www.opencalais.com/documentation/calais-web-service-api/api-metadata

[6] JSON (JavaScript Object Notation) is a lightweight data-interchange format. It is easy for humans to read and write. It is easy for machines to parse and generate. It is based on a subset of the JavaScript Programming Language. For more information see http://www.json.org/

Acquisition	CompanyTechnology	PersonAttributes
IndustryTerm	Alliance	CompanyTicker
PersonCareer	MarketIndex	AnalystEarningsEstimate
CompanyUsingProduct	PersonCommunication	MedicalCondition
AnalystRecommendation	ConferenceCall	PersonEducation
MedicalTreatment	Arrest	ContactDetails
PersonEmailAddress	Movie	Bankruptcy
Conviction	PersonRelation	MusicAlbum
BonusSharesIssuance	CreditRating	PersonTravel
MusicGroup	BusinessRelation	DebtFinancing
PoliticalEndorsement	NaturalFeature	Buybacks
DelayedFiling	PoliticalRelationship	OperatingSystem
CompanyAccountingChange	DiplomaticRelations	PollsResult
Organization	CompanyAffiliates	Dividend
ProductIssues	Person	CompanyCompetitor
EmploymentChange	ProductRecall	PhoneNumber
CompanyCustomer	EmploymentRelation	ProductRelease
PoliticalEvent	CompanyEarningsAnnouncement	EnvironmentalIssue
Quotation	Position	CompanyEarningsGuidance
EquityFinancing	SecondaryIssuance	Product
CompanyEmployeesNumber	Extinction	StockSplit
ProgrammingLanguage	CompanyExpansion	FamilyRelation
Trial	ProvinceOrState	CompanyForceMajeure
FDAPhase	VotingResult	PublishedMedium
CompanyFounded	IndicesChanges	

Table 4.2: OpenCalais Possible Facts and Events

To extract the necessary information, the CalaisParser class in Java is used. It uses a library for the JSON Parser, and then transfers the JSON response into a list of paragraphs. Each paragraph contains one type of information. To illustrate, Listing 4.1 shows an example of event acquisition:

```
typereference :http://s.opencalais.com/1/type/em/r/acquisition
type :acquisition
typegroup :relations
company_acquirer:
http://d.opencalais.com/comphash-1/1ffc29a4-8ab5-3f00-acdd-06f24bc8ad3b
status :planned
company_beingacquired:
http://d.opencalais.com/comphash-1/59f5ab10-5235-3add-bf85-6777c8116b6e
detection :johannesburg at 10 a.m. local time to consider [a takeover
 bid from vale sa.] jinchuan group co. has made a higher offer.
exact :a takeover bid from vale sa.
prefix :johannesburg at 10 a.m. local time to consider
suffix : jinchuan group co. has made a higher offer
```

Listing 4.1: Example of Information Extracted from JSON

After obtaining that list, the existing information is extracted by using regular expressions, and then stored associated to the existing record in the database. The steps of extracting the events are being explained along the example of Listing 4.1. First, the event is identified, which is in this case *acquisition*, and both the *company acquired* and the *company being acquired* are extracted. The companies are written as an URI[7]. The motivation to use URIs instead of plain words lies in the ambiguity of names and language in general: Consider the word ’London’. London is the capital of the United Kingdom, but also a city in Canada and in several states of the USA. Furthermore, several films are called ’London’, as well as various novels and music titles. Thus, using the only the word London without further context does not provide enough information to identify the entity in question. URIs can be retrieved by HTTP response to OpenCalais web linked data.

A further issue of recognising companies’ names is the fact that names can be written in different ways. As mentioned above, the dataset contains a table that consists of all of the possible names of companies and their indices. There should be a match between the names of companies in this table, and the names collected from OpenCalais outcomes. This matching turned out to be difficult and took a long time to be sorted out, because names in OpenCalais contain either the look up symbols, or some other different words or punctuations. For instance, “*corp.* and *corporation*”, “*plc* and *plc.*”, “*limited* and *ltd*”, “*inc.* and *incorporation*”. A script in MySQL was written for the purpose of picking up the matching cases from both and put them into a new table in the database to be used as a final source for the companies.

For information completion, regular expressions are used to extract the publication date of the article, which might be different from the date of crawling the web, as

[7] URI or Uniform Resources Identifier, includes a string of characters for the file-name and may also contain the path to the directory of the file. URIs provide a standard way for resources to be accessed by other computers across a network or over the World Wide Web. They are used by software programs such as Web browsers and P2P file-sharing programs to locate and download files. See: http://www.techterms.com/definition/uri, and http://www.opencalais.com/documentation/linked-data-entities

it is important to specify the stock prices and the index prices of the same date of publication. In addition, the news publishers are extracted, to detect the reliability of the news, and delete any unreliable resources since the study considers formal news only and ignores other news types[8].

4.4.2 CHARACTERISTICS OF FINAL DATA

The news article records contain more than 120,000 news articles from different resources. For daily news retrieval and storage in the database, the web crawler needs a ranging time between half an hour to two hours. On average, the number of stored records is 1000 records per day in the time when the data was first restricted to FTSE100, and 3000 records per day after removing the restriction. Processing the news items, extracting the article and running OpenCalais need around 20 second per record. Parsing of OpenCalais outputs and extracting the rest of the information need 5 seconds per item.

The final result after processing the data through the steps mentioned earlier includes only 17 types of events in 682 news items that are relevant for the study. Fig. 4.1 shows the extracted events and their frequencies among the data where *acquisition_A* indicates the case of company acquirer, and *acquisition_B* indicates the company being acquired. These two cases are treated as two separate events. The companies covered by those events are listed in the following indices or trading markets: Dow Jones Composite, S&P 100, S&P 400 MidCap, S&P 500, Nasdaq Global Select Market, S&P 600 SmallCap, Nasdaq Capital Market Composite, FTSE100, Nasdaq 100, NYSE, AMEX, and DAX. For some companies which are listed in more than one index, the index was chosen randomly from the database.

At this stage, the data set contains a detailed record for each news item. That includes the event it mentions, the corresponding company to this event, the publica-

[8] Other types of news are Pre-News and Social Media, as well as rumours

tion date, the returns of the stocks in the corresponding date, the index in which the corresponding company is listed on, and its return in the same day of publishing the news and in a one day earlier.

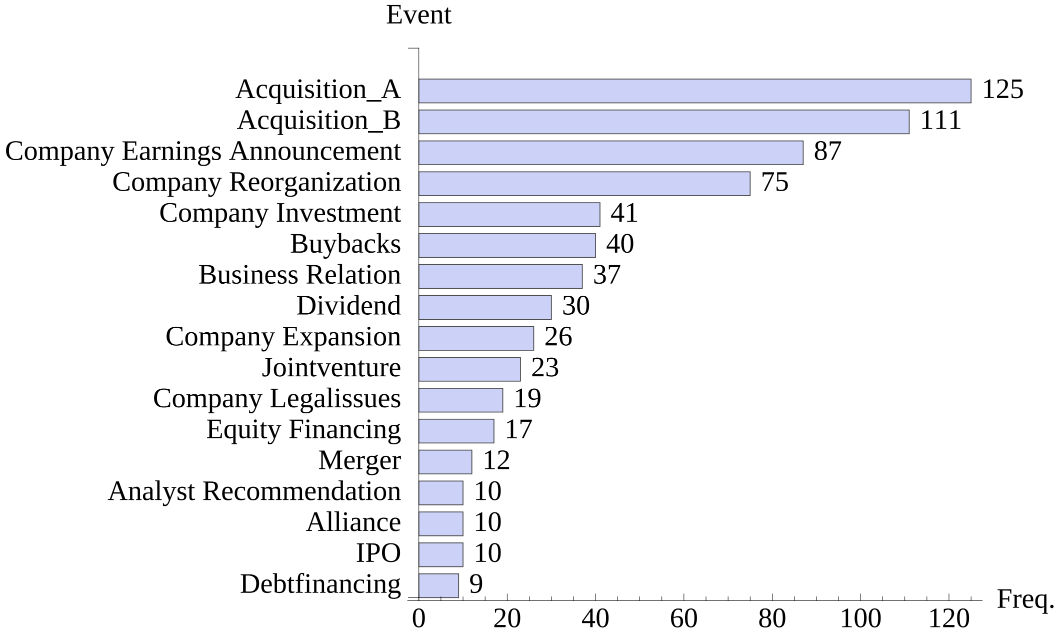


Figure 4.1: Frequencies of Events in the Available Data

4.4.3 THE MODEL

In the Event-Based Sentiment Model, the events that are announced in the news are used to classify the news into positive, negative or neutral categories. The model characterises the relationship between event occurrences and stock returns of the corresponding companies. For each news item, the event is detected and the return on the same publication date is computed. Over the whole data set, frequencies of occurring events, coupled with either positive or negative returns are enumerated. Afterwards, various statistical methods that inspect the significance of the relationship between event occurrences and returns are utilised. If the event is significantly coupled with positive returns, then any news item that contains this event will be assigned a positive sentiment. On the contrary, if the event is significantly linked to negative returns, the associated news item is then assigned a negative sentiment. Moreover, for events with no impact on the returns, the related news items will be assigned a neutral sentiment. In this light, the following experiments assess first the relationship between event occurrences and stock market returns. Based on the results, the model can be built.

4.4.4 Experiments and Results

At first, the Naive Bayes method is employed to examine whether event occurrence has an effect on the daily stock market returns. To establish this relationship, the probability of positive returns conditional on event occurrence is calculated for each event.

Let $\{r_t\}_{t=1}^{T}$ denote the daily returns over a time period T. Returns are calculated based on the difference between opening and closing prices of time t. Let $E = \{e_1, e_2, \ldots, e_n\}$ denote a set of the possible events that might occur, and e_n^t is a binary variable denotes the occurrence of event n in time t

$$e_{n,t} = \begin{cases} 1, & \text{the event } n \text{ occurred} \\ 0, & \text{else.} \end{cases} \tag{4.4.1}$$

Then $\mathbb{P}(r^+|e_n)$ denotes the probability of positive returns, given an event n has occurred:

$$\mathbb{P}(r^+|e_n) = \frac{\sum_{t=1}^{T} e_{n,t}\, r_t^+}{\sum_{t=1}^{T} e_{n,t}}, \tag{4.4.2}$$

where $r_t^+ \in \{1, 0\}$ depends on whether the return is positive or not.

The values of $\mathbb{P}(r^+|e_n)$ for all events are illustrated in Fig. 4.2.

Fig. 4.2 shows that several events correspond well to positive returns. For event "alliance" the probability of positive return given the occurrence of this event, raises attention as it is 82%. One would consider the very low percentages also as interesting results, since they represent a high probability for the returns to be negative

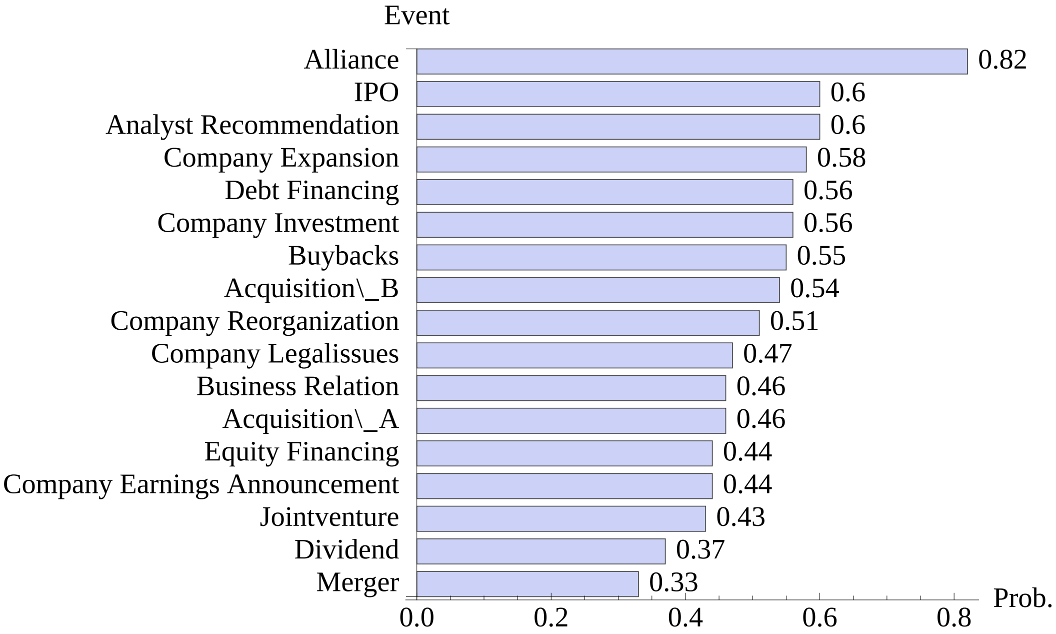


Figure 4.2: Conditional Probabilities of Rising Returns

when the events occur, because

$$\mathbb{P}(r^-|e_n) = 1 - \mathbb{P}(r^+|e_n). \tag{4.4.3}$$

Consider for instance the event “merger”, $\mathbb{P}(r^+|e_{\text{merger}}) = 0.33$ implies that $\mathbb{P}(r^-|e_{\text{merger}}) = 0.67$. That indicates that given that the event is “merger”, the probability that the returns is negative is 67%.

The naive Bayes method gives a first indication of event candidates that could influence the returns. However, to properly assess the relationship between returns and event occurrences, a more intricate analysis is necessary. For some of the events it is clear that no effect is measurable. One example is “company reorganisation”. $P(r^+|e_{\text{comp.reorg.}}) = 0.51$ and accordingly $P(r^-|e_{\text{comp.reorg.}}) = 0.49$ shows clearly that this event does not have any measurable effect on returns. A different case is the event “alliance”, with $P(r^+|e_{\text{alliance}}) = 0.82$. However, the question that will be answered in the following is whether this shift from 50-50 chances is statistical significant. In order to assess this, the t-test will be used where applicable. If

	eventname	pvalues
1	acquisition_A	0.0008
2	acquisition_B	0.00007
3	alliance	0.45
4	analystrecommendation	0.80
5	businessrelation	0.37
6	buybacks	0.47
7	companyearningsannouncement	0.000
8	companyexpansion	0.007
9	companyinvestment	0.12
10	companylegalissues	0.04
11	companyreorganization	0.00002
12	debtfinancing	0.49
13	dividend	0.098
14	equityfinancing	0.81
15	ipo	0.17
16	jointventure	0.58
17	merger	0.14

Table 4.3: Results of Jarque-Bera Test

the data does not meet the parametric assumptions, then non-parametric Wilcoxon Rank test will be used. This separation is made in order to use the higher power of parametric test where possible and use the weaker non-parametric test only where it is necessary.

The null hypothesis of the t-test is that the sample mean is zero. The test can only be performed when the testing sample follows a normal distribution. To assess the normality, the Jarque Bera (JB) test [19] will be employed in the following. The JB-test tests under H_0 whether the sample is normally distributed. The results for all return samples that correspond to respective events can be found in Table 4.3.

Table 4.3 indicates that for events alliance, analyst recommendation, business relation, buybacks, company investment, debt-financing, dividend, equity-financing, IPO[9], joint-venture and merger the null hypothesis cannot be rejected since the p-values of the test are all larger than 0.05. Hence, the t-test will be used those events.

[9] Initial Public Offering

As for the rest of the events where each of the p-values is less than 0.05, the null hypothesis can be rejected, and a non-parametric test, the Wilcoxon signed-rank test, will be used for those events.

RESULTS

The tests results are shown in Table 4.4. Only for event "alliance", a significant positive shift of returns can be found. This is in line with the Naive Bayes results in the previous part. For all other events, each of the p-values is larger than 0.05 which means that the null hypothesis that the means are zeros, cannot be rejected, and the mean does not have the value zero. Therefore, the announcements of the events: analyst recommendation, business relation, buybacks, company investment, debt-financing, dividend, equity-financing, IPO, joint-venture, merger, acquisition for both companies acquired and being acquired, company expansion, company legal issues, company reorganisation, and company earnings announcement, do not have a significant effect on the stocks returns on the corresponding days to those events.

REGRESSION MODEL

After the existence of the effect of event "alliance" has been established in the previous experiments, the extent of this effect will be measured in the following regression model. The model assumption is that the stock price returns are affected by the returns of their corresponding market indices in the previous day, and by the occurrence of the event. As mentioned earlier, for stocks that are parts of more than one index, only one index then is chosen randomly from their index list.

The regression problem can be represented by:

$$\min_{\beta_0,\beta_1,\beta_2} \left\{ \sum_{i \in N} \left[r_i - \beta_0 - \beta_1 I_i - \beta_2 e_{i,\text{alliance}} \right]^2 \right\}, \tag{4.4.4}$$

event	p-value	test
alliance	0.02	t-test
analystrecommendation	0.32	t-test
businessrelation	0.44	t-test
buybacks	0.25	t-test
companyinvestment	0.68	t-test
debtfinancing	0.44	t-test
dividend	0.61	t-test
equityfinancing	0.98	t-test
ipo	0.46	t-test
jointventure	0.89	t-test
merger	0.84	t-test
acquisition_A	0.27	Wilcoxon
acquisition_B	0.85	Wilcoxon
companyearningsannouncement	0.25	Wilcoxon
companyexpansion	0.30	Wilcoxon
companylegalissues	0.68	Wilcoxon
companyreorganization	0.35	Wilcoxon

Table 4.4: Results of t-test and Wilcoxon signed rank test

where N denotes the set of news items r_i denotes the corresponding stock return to news item i, and I_i denotes the corresponding index return to news item i in the previous day, while $e_{i,\text{alliance}}$ denotes the binary variables of the event alliance occurrence.

The parametrised regression equation for the event “alliance” becomes:

$$r_i = -0.0002 + -0.172 I_i + 0.031 e_{i,\text{alliance}}. \tag{4.4.5}$$

However, the value of $R^2 = 0.022$ is so small, indicating that the variations of the explanatory variables are able to explain only 2.2% of the variances of the returns. Also the Jarque Bera test, that assesses the normality of the residuals, shows that the residuals are not normally distributed and therefore not a white noise. This means that the model does not fit the data, and the t-test statistics cannot be interpreted reasonably because the main assumption is violated.

At this point in the analysis, an interesting argument arises. The frequency of the event “alliance” is low, so it is likely that the effect which was concluded in the regression model for this event is solely based on the index out-performance on the same day of the “alliance” occurrence. In other words, the reason for finding an effect, or even finding high probability of rising prices given this event has occurred, might be a strong performance of the index, since it is likely for the stock prices to rise when the index value is rising. This linear dependence is simply rooted in the way a stock market index is calculated. Another argument takes the market efficiency hypothesis in consideration: the information known about the index has an effect on the stock prices in the same day [101]. To test the credibility of this argument, the model will now be tested on abnormal returns. The abnormal returns are calculated by deducing the index return of the same day from the returns of the stocks [101]. This clears the returns from the noise of the market environment. The results of the new model are in Table 4.5.

	Estimate	Std. Error	t value	Pr($>|t|$)
(Intercept)	0.0028	0.0013	2.14	0.0329
index_return_p	-0.2704	0.0670	-4.04	0.0001
event_alliance	0.0136	0.0103	1.32	0.1875

Table 4.5: Results of Regression on Abnormal Returns for event Alliance

The results suggest that there is no longer an effect for the occurrence of event “alliance” after the out-performance of the index in the same day is removed. Then by testing the residuals for the return of the index in the previous day, the Jarque Bera test proves the non-normality of the residuals, and the model does not describe the data well.

To conclude, the results in this section illustrate that most of the investigated events do not have an impact on the stock market returns. In fact, only the event “alliance” is a candidate. However, the extent of this effect could not be measured with a linear model. It is possible that a non-linear model or a model with more variables is able to measure the effect, but this analysis is beyond the scope of this thesis. The next section investigates the investor sentiment and its effect on the stock market returns.

4.5 Reader-Based Sentiment Analysis

In this section, a new model to classify news sentiment is proposed. It uses an automated method to label the news and then utilises the framework that is suggested in Chapter 3 to test the model predictability. First the news is labelled as positive or negative for training purposes, then the classification process is proceeded, and the final predictions are validated.

4.5.1 Labelling the news

As mentioned earlier in this thesis, in sentiment analysis, there is another phase next to the opinion holder or writer: the opinion reader. The reader's opinion perspective, to the best of my knowledge, has not been searched before in the literature. Financial news has the prospect of capturing the reader's opinion. The reader in this respect is the investor in the stock market who reads the news and translates it into a buy, sell, or hold decision. This decision reflects what the investor thinks as well as what is the investor sentiment about the news. Accordingly, the reader sentiment can be captured by looking at the stock market returns which represent how the news are reflected in the market.

A new method, the Reader Sentiment Indicator method (RSInd), is suggested for labelling the news items. This method is explained as follows. First, for each news item i the return $r_i^{(t)}$ in the publication date t is calculated as in the previous section. To make sure that the stock return is not affected by the strong performance of the marketplace, such as in the case of event ”alliance”, abnormal returns are computed. Each news item is then labelled positive or negative based on the abnormal returns of the corresponding stock in the day the news is released. To endorse the news effect capture, a threshold for the abnormal return needs to be defined to enhance the labelling accuracy. For example, if the threshold is set at 5%, the news would be labelled as positive if the abnormal return is above 5% and negative if the abnormal return is below -5%. This is done in order to be more certain about the news sentiment, and to allow us to exclude cases where the news is neutral.

By applying RSInd on the 682 news items data, 192 news items are picked and labelled, from which 84 items are negative, and 108 are positive. A set of 84 positive items is randomly selected to be used in the experiment to create an equally distributed set. Therefore the final data set contains 168 items, of which 84 items are positive, and 84 are negative.

4.5.2 Refining the Data

The suggested framework in Chapter 3 is utilised for data pre-processing. The $F^{(f)}$ matrix is constructed according to different χ^2 values (0.05, 0.01, 0.001, 0.005), and using the feature presence (FP) weighting method. Fig. 4.3 illustrates the relationship between realisations of the χ^2 statistic and the frequency $F_i^{(t)}$ over all documents for each feature. Obviously, features in the upper-right corner of these panels are most desirable, as one wants to find a sufficiently well predicting feature that is frequently found. However, the panel on the left-hand side shows that this combination is rarely found. Most of the features spread over the lower left-hand side: low explanatory power (χ^2) coupled with low frequencies ($F_i^{(t)}$) . There are a number of features that have a statistical significant explanatory power. These are illustrated in the right-hand side panel. Curiously, these features resemble many words that a human reader would as well connect to a strong negative sentiment, such as victim, corrupt, violate, etc..

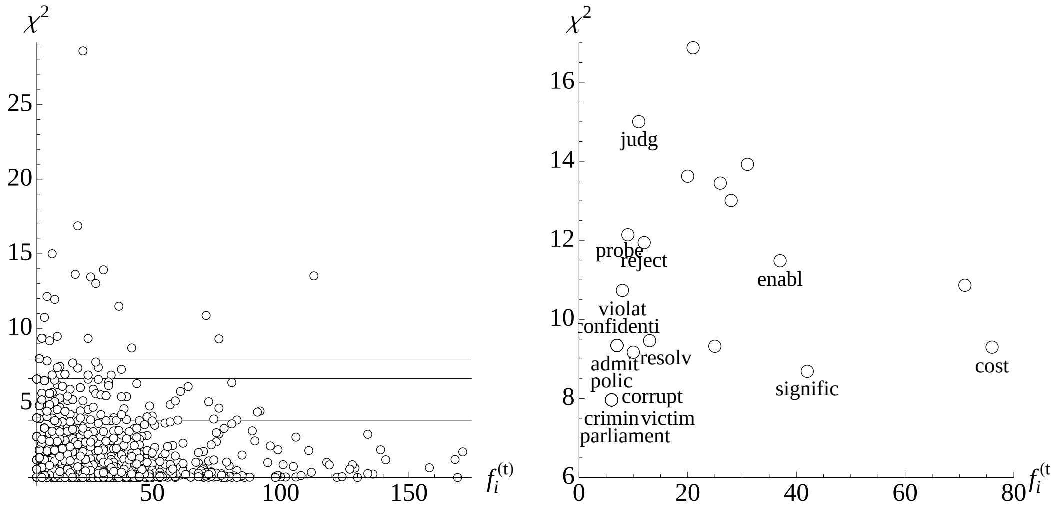


Figure 4.3: The scatter plot illustrates the relationship between the frequency of the features f_i and the features' significance in the classification procedure, measured by the χ^2 test statistic. The three vertical lines in the left hand side correspond to different p-values, starting from the lowest line $p = 0.05(\chi^2 = 3.84)$,$p = 0.01(\chi^2 = 6.63)$,$p = 0.005(\chi^2 = 7.88)$. Most desirable are features in the upper-right corner. They appear in the right hand side graph in a zoomed view

4.5.3 Experiment and results

The SVM classifier is used after constructing $F^{(f)}$. The Gaussian radial basis kernel function is chosen, with the parameter γ that controls for the area in which the support vector has an effect in the data space. The parameters C and γ are chosen through a grid search due to the sensitivity of SVM performance to their values. The data is divided into two parts: one for training and the other for testing, by ratio $9 : 1$, that is $9/10$ parts were used for training and $1/10$ for testing. Then training is performed with 10 fold cross validation for classification. The experiments are constructed based on four different p-values of the χ^2 test. They are: (0.05, 0.01, 0.005, 0.001). Each experiment is run for 60 iterations. The average results are shown in Table 4.6.

Table 4.6 shows the effect of the p-value on the classification performance. The table illustrates that the relationship of average accuracies and the p-value is somewhat peaked. Accuracies at the lower and upper boundary of the investigated interval are less than the accuracies found in the middle of the interval. This finding

p-value	features	Accuracy	Precision	Recall	F-Measure
0.05	257	75.78	69.67	79.72	74.15
0.01	51	76.22	71.05	79.73	74.84
0.005	31	75.47	70.18	78.91	74.05
0.001	12	73.95	71.99	75.26	73.40

Table 4.6: Averaged classification accuracies in percentages after using χ^2 with different critical p-values and different numbers of features. Averages are computed over 60 iterations.

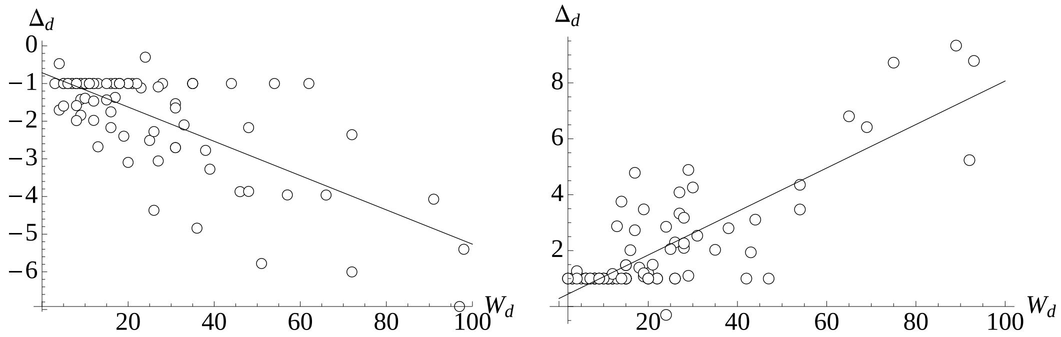


Figure 4.4: Scatterplots depicting the relationship between the number of features W_d and the distance to the separating hyperplane Δ_d for each document in the negative category (left hand side) with regression line $\Delta_d = -0.678 - 0.021W_d$ ($R^2 = 0.167$). For the positive class (right hand side) with regression line $\Delta_d = -0.559 + 0.094W_d$ ($R^2 = 0.702$)

suggests that there exists some optimal p-value that leads to the highest performance. However, within the bounds of this investigation, only local optimality can be stated, as numerical investigations are computationally extensive.

Investigating further in the classifier performance, likewise in the experiment on movie reviews data, the relationship between the number of features in each document and the distance to the separating hyperplane Δ_d is inspected. The results are presented in Fig. 4.4.

Fig. 4.4 allows to draw the same conclusion as in the movie reviews section. The figure shows a weak relationship between the number of features in the documents and their distance to the separating hyperplane in the negative category. However, the graph shows a positive relationship between the number of features in the documents and the distance to the separating hyperplane in the positive category. It is stated that the confidence in predicting positive classes increases with the number of features in the document, contrary to predicting negative documents.

4.5.4 Model Validation

To test whether the reader-based sentiment model can be used for a trading strategy, a trading strategy based on the model is tested against a random strategy. To conduct this test, three different portfolios are constructed. The stocks in each portfolio are sampled from the data set that has been used throughout this section. To be more precise, the whole sample consists of news-returns pairs from the testing set. For the sake of simplicity, it is assumed that the investor decides to buy stocks depending on the news sentiment and hold them for one day. Then, portfolio_P contains all the pairs for which the news item has been predicted as positive in the reader-based sentiment model. Portfolio_N contains all the pairs with negatively predicted news items. The random portfolio, portfolio_R, contains the entire set of pairs, regardless from the sentiment prediction of the news items. Whereby the reader-based sentiment model is used to categorise the news items. The returns distributions of portfolio_P and portfolio_N and portfolio_R are shown in Fig. 4.5.

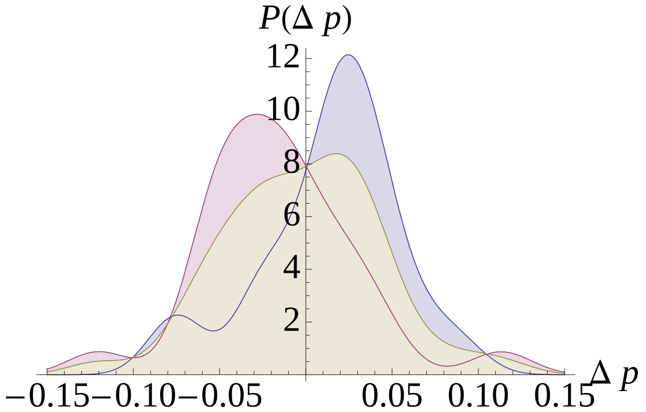


Figure 4.5: The distributions of daily returns in the different portfolios. The red graph shows the daily returns distribution of portfolio_N. The blue graph shows the daily returns distribution of portfolio_P. The yellow graph indicates the returns of portfolio_R.

To test if the reader-based strategy can generate better returns than a random strategy, the Wilcoxon signed rank test is used. It tests whether there is a significant difference between each of the red and blue distribution and the random strategy distribution. The test show a significant difference between the expected return in portfolio_P using the reader-based sentiment strategy, and the expected return using the random strategy in portfolio_R (p-value= 0.03). The test reveals the same results for portfolio_N using the reader-based sentiment strategy (p-value= 0.008). The conclusion here is that the reader-based sentiment strategy is significantly better than the random strategy, for both short and long positions.

These differences can be quantified. Using the reader-based sentiment model for a trading strategy, the probability that a predicted positive return is indeed positive is 69% with an expected daily return of 1.39%. Furthermore, the probability that a predicted negative return is indeed negative is 67% with an expected daily return of -1.5%.

4.6 CONCLUSION

In this chapter, two novel approaches toward financial news sentiment are explored: event-based and reader-based sentiment analysis. Event-based sentiment analysis detects the sentiment in the news according to the events that are announced in them. In order to achieve that, the relationship between stated events and the stock returns of the corresponding company have been inspected. To comprehend this relationship, different parametric and nonparametric statistical methods have been employed. A simple Naive Bayes approach shows a strong relationship between event “alliance” and the returns. Further tests were then carried out to measure whether specific events has a statistical significant effect on the stock price. Those tests confirm the outcome of Naive Bayes approach: “alliance” event shows a significant effect on the returns. A regression model is used to inspect the extent of this effect. After reducing the noise that originates from the market’s activities, by using abnormal returns, the significance of the event “alliance” effect could not be re-established in a linear model.

Therefore, the proposed event-based model cannot be built in this case because no relationship could be found between almost all of the events and stock market returns. This conclusion can be data specific. Thus, the possible success of the event-based sentiment model cannot be ruled out yet. A bigger data set with higher frequencies of event occurrences could show a pattern that connects the events to stock market returns. Also, a non-linear model might explain the dynamics of this relationship. Another possible technique is to incorporate economic expert opinions about events that have an impact on stock markets in the model. To assess those options, more experiments on bigger data sets are to be carried out in the future.

As for the reader-based sentiment analysis, the model detects the reader sentiment based on the stock market returns. It is argued in this thesis that the reader sentiment in financial news is reflected by the buy or sell decisions in the market, or in other words in the stock market returns. A method to detect the reader sentiment and label the news accordingly is suggested (RSInd). To test the validity of this approach, SVM is used to classify the sentiment of the news. Afterwards, the predicted documents are used to test a trading strategy that uses this model against random strategy. As a result, the trading strategy that is built based on this model outperforms the random strategy significantly. This suggests that the reader-based sentiment model is a valid model for labelling the news and predicting their sentiment.

CHAPTER 5

CROSS DOMAIN SENTIMENT ANALYSIS

5.1 INTRODUCTION

In the previous two chapters, two different types of texts were analysed: movie reviews, and financial news. Both types were subject to the classification process that was proposed in Chapter 3. However, the results were significantly different. The classifier predicted movie reviews classes with a accuracies up to 94% (see Fig. 3.6). On the other hand, the accuracies in the financial news data were lower, ranging from 70% to just above 79%. The main reason behind this difference is that movie reviews cover only one domain, while the financial news covers a wide variety of domains. For instance, news about a law suit, covers the “legal” domain, while earning announcements or dividends are parts of “accounting” domain, etc.. Thus, the training data might not provide a sufficient coverage for each of those different domains. This problem is addressed in cross domain sentiment analysis.

The cross domain sentiment problem is stated as follows: given a set of labelled data from one domain (the source domain), and a set of unlabelled data from another domain, (the target domain), how can documents in the target domain be classified using a trained classifier on the source domain?

As mentioned earlier in Chapter 2, most studies in this branch of sentiment analysis tackle this problem by detecting domain independent words/features to bridge between the two domains. In other words, those features are used to transfer the sentiment from the source to the target domain. In this chapter, different methods to detect domain independent features are explored separately and in combination. Finally, those methods are used in different cross domain classifiers.

The chapter is organised as follows. In Section 5.3, domain dependent and independent features are investigated. A new method to detect domain independent features based on their frequencies and their mutual information is proposed. In Section 5.4, different methods for cross domain sentiment classifications are investigated and compared against each other. After that, a lexical method for sentiment classification is evaluated in Section 5.5. Section 5.6 closes with concluding remarks.

5.2 DATA SETS

The results of the event-based model in the previous chapter show that the available financial data is not varied enough in terms of events, or in other words in terms of topics/domains. Furthermore, the sample of each event is not representative, and does not allow to use it in cross domain sentiment as it not large enough to construct training and testing sets for classification purpose. Thus, in this chapter, the cross domain methods are applied on other data sets.

Five different data sets are used in this chapter. These data sets comprise different domains: movies, music, computers, hotels, and books. Each set contains 50 items that are distributed equally between negative and positive category. All data sets are publicly available[1], and they are part of the SFU Review Corpus. They were

[1] http://www.sfu.ca/ mtaboada/research/nserc-project.html

collected from the Epinions website[2], and were first used in [111, 110].

For each of those data sets, the following procedure is applied. Employing the methodology from Chapter 3, Section 3.1, the matrices $F^{(b)}$ and $F^{(f)}$ are constructed. Two weighting methods are used, feature frequencies (FF), and feature presence (FP). $F^{(b)}$ is the feature frequency or feature presence matrix before any pre-processing, and $F^{(f)}$ is the feature frequency or feature presence matrix after filtering.

5.3 Domain (In-) Dependent Features

Domain dependent features are the words that are used exclusively in a certain domain. Some of those features are non-informative from a sentiment classification perspective. They have high frequencies in all documents and do not relate to the sentiment category. The removal of those words from the text helps reduce the text dimensionality and hence improves the classification performance (see Chapter 3). For example, the words *movie, actor, play, role* are movie domain specific, in which they occur in most of the documents with high frequencies, but they are not informative in terms of the document's polarity. They will be referred to as domain specific features in the course of this chapter.

However, some other domain dependent features are important for sentiment classification. In this context, they are words that have certain polarities that drive the documents polarities. For example, in monitor reviews, the features *thin, high resolution, crisp screen* are domain dependent, and they have a positive polarity. On the contrary, they will not appear in a book review. This type of features will be referred to as domain dependent features.

Domain independent features are words that are not specific for any domain. These

[2] http://www.epinions.com/

features appear in different domains in high frequencies and have the same effect on the text orientation [86]. This property makes them important in cross domain sentiment classification, because it makes it possible to use them to transfer the sentimental knowledge from the source to the target domain. This is accomplished by exploring their relations to co-occurring features in the same document/sentence of the source domain. In this thesis, several methods are used to detect domain independent features, some of which are new.

5.3.1 Feature Frequencies Method

This new method is used to extract domain dependent and independent features. It solely relies on the feature frequencies in the data sets. It employs the fact that domain dependent features appear in similar frequencies in all categories. A list of all features in all documents and their frequencies over positive and negative categories is constructed from $F^{(b)}$. An example of this list for movie data is shown in Table 5.1. Then, features that appear only in one category are removed. The proposed method utilises this list to select features that has almost equal frequencies in both categories, and labels those features as domain dependent. The remaining features are labelled as domain independent. This is illustrated in the following.

Firstly, another truncation is applied on the list. Only features that appear in $F^{(f)}$ are selected. This is to choose features that are significant to the text polarity. In this case, all domain specific features will be removed. Secondly, to split domain dependent and domain independent features in the new list, a cutoff between the sum of frequencies over both categories, and difference in frequencies over both categories is chosen. These are the last two columns in Table 5.1. The cutoff is defined by the value that minimises the absolute difference of frequencies counts while it maximises the sum of counts over both categories. Fig. 5.1 illustrates this tradeoff.

	feature	Negative counts	Positive counts	Absolute diffs. counts	Sum of counts
1	movi	603	580	23	1183
2	time	493	526	33	1019
3	more	495	523	28	1018
4	thei	520	490	30	1010
5	charact	461	500	39	961
6	make	457	493	36	950
7	plai	420	421	1	841
8	even	431	409	22	840
9	onli	426	393	33	819
10	good	378	425	47	803
11	stori	345	413	68	758
12	first	351	385	34	736
13	come	348	377	29	725
14	director	375	345	30	720
15	well	325	392	67	717
16	take	332	376	44	708
17	much	319	384	65	703
18	year	332	369	37	701
19	look	374	321	53	695

Table 5.1: An example of features frequencies over both categories, using $F^{(b)}$ in Dat-1400. The negative counts and positive counts columns are the number of occurrences of features in the negative and positive documents respectively. The absolute diffs. of counts column is the value of the absolute difference in frequencies between the negative and the positive categories, while the last column is the sum of frequencies over both categories.

Lastly, all features with frequencies above the cutoff are labelled domain dependent. The remaining features after choosing the domain dependent features are labelled as domain independent. In addition, the features that appear only in one category, that were excluded earlier, are labelled as domain independent. This feature frequencies method is applied on all of the data sets, and the lists of domain dependent and independent features are stored for later use in the cross-domain classification models.

5.3.2 Mutual Information

Mutual information (MI) is a feature selection method used to measure the mutual dependence between the features and the categories. Let $c_i \in \{pos, neg\}$ represent

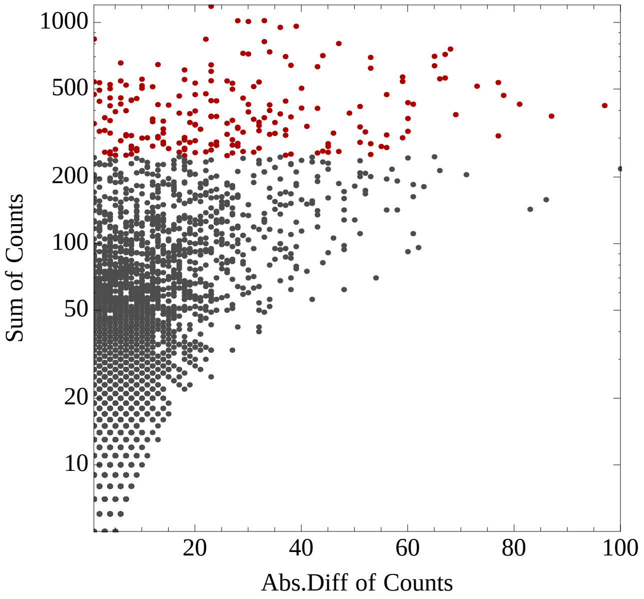


Figure 5.1: The figure illustrates the tradeoff between the sum of frequencies and difference of frequencies to choose domain specific features. The chosen features are represented by the red points.

the set of categories, and $f_j \in \{f_1, f_2, \cdots, f_n\}$ the set of features in all documents. n is the number of features in all documents, and N is the number of documents. In addition, $\Pr(f_j, c_i)$ is the probability that the feature f_j occurs in category c_i. $\Pr(f_j)$ is the probability that in a randomly chosen document, the feature f_j exits:

$$\Pr(f_j) = \frac{\text{\# occurrences of feature j in all documents}}{\text{\#occurrences of all features}}. \tag{5.3.1}$$

$\Pr(c_i)$ is the probability that a randomly chosen document belongs to category c_i. Then, the mutual information between the feature f_j and the category c_i is computed by:

$$MI = \Pr(f_j, c_i) \cdot \log_2 \left(\frac{\Pr(f_j, c_i)}{\Pr(c_i) \cdot \Pr(f_j)} \right). \tag{5.3.2}$$

The higher the value of MI is, the higher the mutual dependence between the feature and the category. This becomes especially clear when a limiting case is considered: Consider two random variables X and Y. If X and Y are independent, then $\Pr(x, y) = \Pr(x)\Pr(y)$ and thus $MI = 0$. Furthermore, the MI score measures

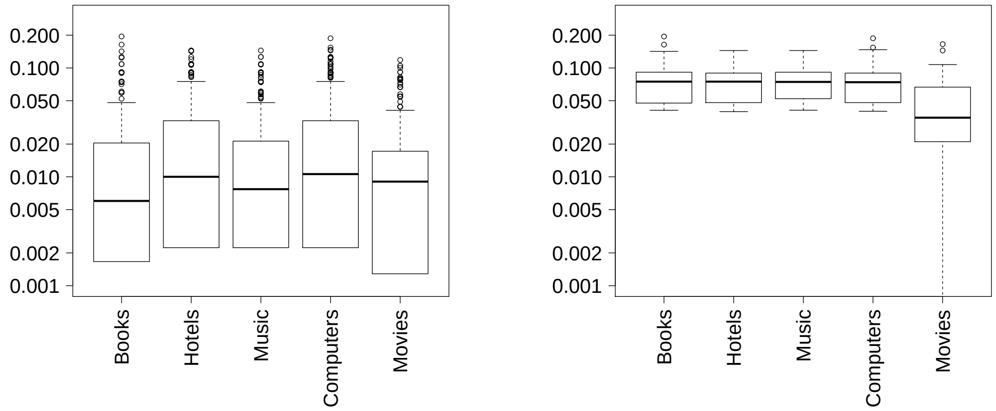


Figure 5.2: The box plot on the left hand side shows the MI range on all features, while the box plot on the right hand side shows the MI range on the selected significant features. In the right hand side figure, the features are significant to the categories, the MI range is shifted to the top

how much uncertainty reduction there is about f_j, after gaining knowledge about c_i. In other words, how much information f_j conveys about c_i. Thus, the score can also be interpreted as domain dependency: features with high MI score are domain dependent.

The MI method is applied on all the data sets. The outcome is a list of features, where features with high MI score are labelled as domain dependent, while the features with a lower MI scores are labelled as domain independent. Afterwards, same strategy that is used earlier to exclude domain specific features from the list by selecting only the features that appear in $F^{(f)}$, is used. The box plots in Fig. 5.2 demonstrate the difference in the ranges of MI values before and after using this strategy, in other words, before and after employing the $F^{(f)}$ matrix.

5.3.3 In-domain Selection

After obtaining the lists of domain dependent and independent features using the two previous methods, the lists are compared to each other. The features that appear in the outcome of both methods are selected. They are then used as the final sets of domain dependent and independent words.

5.3.4 Cross Domain Selection

In cross domain sentiment analysis, no information about the polarity of the documents in the target domain is available. Therefore, the computations of both of the two previous methods cannot be used to extract domain dependent or independent features in the target domain. However, domain independent features are assumed to appear in both, the source and the target domain frequently and have the same effect on the polarity [86]. Because of that, it is possible to detect independent features in the target domain by selecting all features that appear in the target domain, and in the list of independent features of the source domain in the same time.

5.4 Cross-Domain Classification

5.4.1 Baseline Prediction (BL)

In this subsection, the methodology in Chapter 3 is applied. The SVM classifier is trained on the $F^{(f)}$ matrices of each domain, and then used to predict the classes of the remaining domains. The results are shown in Table 5.2.

As it is clear from Table 5.2, the SVM performance is poor comparing to in-domain classification performance. In the following, two different models to infer sentiment from one domain to another are investigated and evaluated against each other and against the baseline prediction. A new evaluation metric is added to the analysis, that is, the true negative rate. It is the accuracy of negatively predicted documents. In the table, it is denoted by Neg Rate and computed by:

$$TrueNegativeRate = \frac{tn}{tn + fp}. \tag{5.4.1}$$

This measure is added for later comparisons between predictions of negative and positive categories for all of the presented methods for cross domains.

Source domain	Target domain	Accuracy	Precision	Recall	F.measure	Neg Rate
music	computers	70	80	66.66	72.72	73.34
books	computers	66	68	65.38	66.66	66.62
movies	computers	70	70	70.00	70.00	70.00
hotel	computers	74	64	80.00	71.11	68.00
music	books	60	60	60.00	60.00	60.00
computers	books	56	28	63.63	38.38	48.37
movies	books	50	40	50.00	44.44	50.00
hotel	books	50	36	50.00	41.86	50.00
music	movies	60	48	63.15	54.54	56.85
computers	movies	58	56	58.33	57.14	57.67
books	movies	54	44	55.00	48.88	53.00
hotel	movies	52	42	52.63	45.45	51.37
music	hotels	62	64	61.53	62.74	62.47
computers	hotels	62	52	65.00	57.77	59.00
books	hotels	60	48	63.15	54.54	56.85
movies	hotels	50	100	50.00	66.66	50.00
computers	music	58	48	60.00	53.33	56.00
books	music	56	73	55.00	62.85	57.00
movies	music	55	60	54.54	57.14	55.46
hotel	music	54	40	55.55	46.51	52.45

Table 5.2: The evaluation metrics for SVM performance on the 5 different data sets. The SVM classifier is trained on the source domain in the first column, and then used to predict the target domain in the second column. Average accuracies are: 58.85, 56.05, 59.97, 56.63, and 57.72 for accuracy, precision, recall, F-measure, and negative rate respectively.

5.4.2 Inferring Sentiment From MI

This model relies on the mutual information values of features in the source domain. It is referred to as the MI model in the course of this chapter. To illustrate this model, one example from the data is used. Let the hotels data set be the source domain D_{src}, and let the movie data set be the target domain D_{trg}. Further, let the outcome from the feature frequency and mutual information methods on the hotels data be denoted by f_{indep} for domain independent features, and by f_{dep} for the domain dependent features. First, the D_{src} is pre-processed, and the $F^{(f)}$ matrix is constructed to show the significant features for the classification. Afterwards, a cross domain selection for independent features in D_{trg} is applied. Once the independent features in D_{trg} are specified, the mutual information score is computed for each feature in D_{src}. The outcome then is a mutual information matrix $M_{i,j} \subseteq \mathbb{R}^{m \times m}$, where the (i, j)th entry is the mutual information value between feature i and feature j, and m is the number of features in the source domain.

One possible approach to analyse the information dependencies in $M_{i,j}$ is to construct a Mutual Information Relevance Network [20]. This network is illustrated in Fig. 5.3

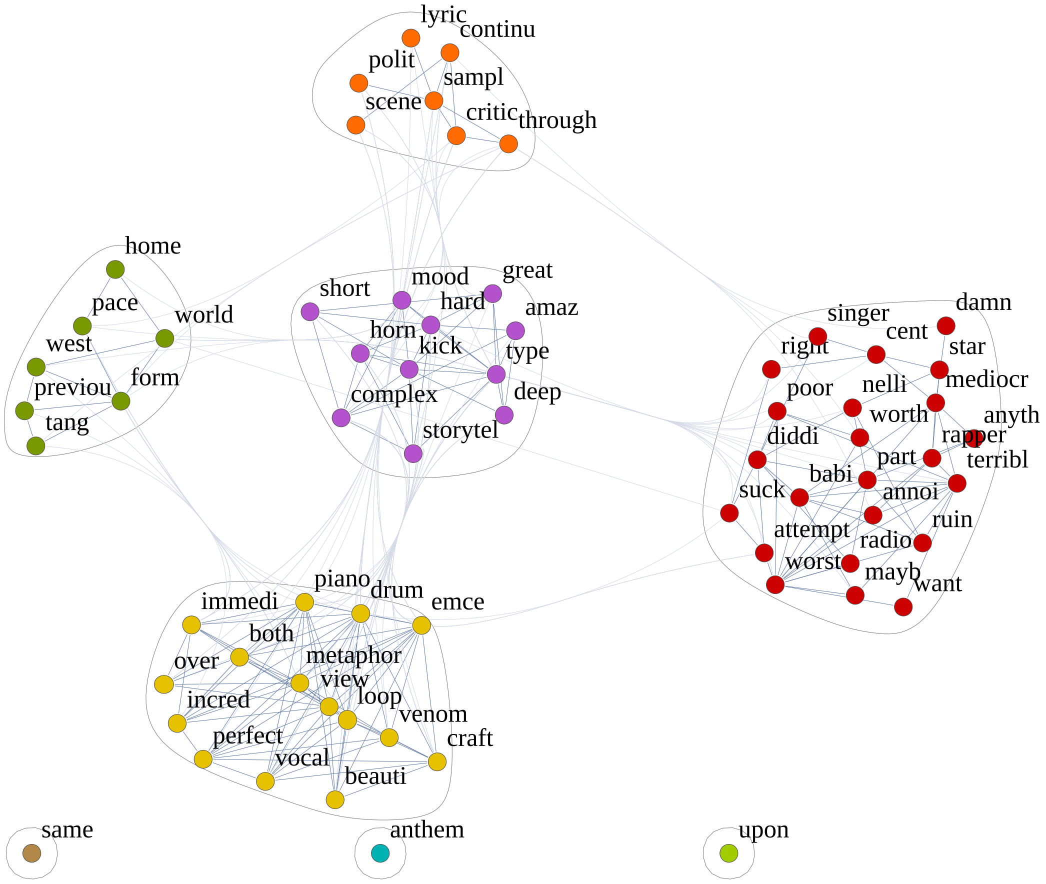


Figure 5.3: The illustration of the mutual information matrix $M_{i,j}$. Features within one group have a higher dependency on each other comparing to other features from other groups.

The clusters in Fig. 5.3 are constructed from the mutual information matrix that has been introduced above. They will be referred to by relevance networks. Every node in the network represents a feature, and every edge between two nodes expresses the mutual information of these two features. If M_{ij} is the (i, j)th entry of the mutual information matrix, then the corresponding adjacency matrix to construct the network is

$$A_{i,j} = H(M_{i,j} - x^*). \tag{5.4.2}$$

$H(x)$ is the standard Heaviside function and x^* is the threshold mutual information (TMI). If $x^* = 0$, then the network is a fully connected graph. A simple measure to summarise the structure of a network is the degree distribution $f(k)$. That is, the probability that a randomly chosen node has k neighbours. Define further, $F(k) = \int_0^k f(v)dv$ as the cumulative distribution function (cdf). Fig. 5.4 shows the change of the structure for different values of x^*. It is obvious that the decay of $1 - F(k)$ has to be faster for higher values of x^*, since the number of edges in the network is decreasing with increasing values of x^*.

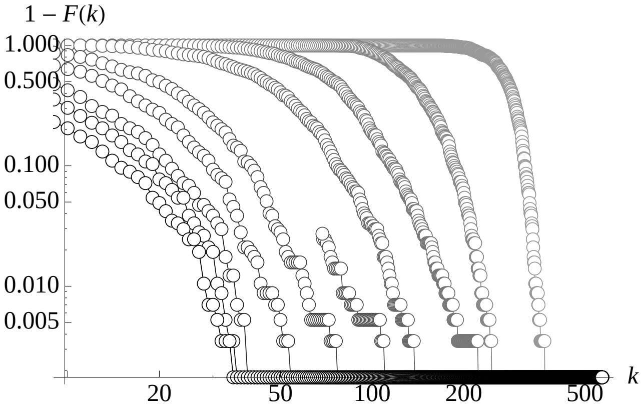


Figure 5.4: Different x^* values and the corresponding cdfs.

However, another observation that can be made in Fig. 5.4 is that the distribution appears similar for $x^* \in [0.1, 0.6]$ before the structure changes drastically. It should be highlighted that a robust analysis of this similarity is not possible in this context because of the compressed domain of $F(k)$. Therefore, the choice of this critical threshold is made heuristically. The following analysis that was first introduced in [20] is conducted to aid the choice process of x^*. Along with the previous example, a range of TMI values is defined for the hotels data set. To select the best TMI, the relationship between TMI and the number of relevance networks is investigated. Fig. 5.5 shows this relation for the hotels data set.

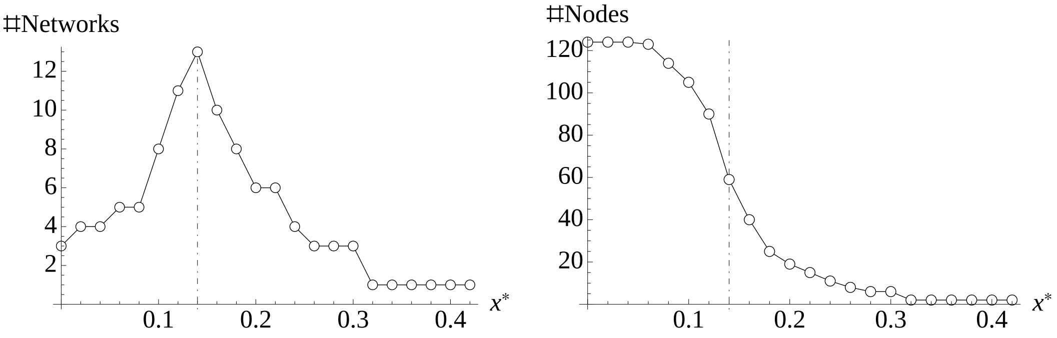


Figure 5.5: The TMI effect on the number of relevance networks and the number of nodes in the all networks. The left-hand side of the graph shows a peak in the number of relevance networks at a TMI value of 0.14. Lower TMI values beyond 0.14 introduce more nodes to the relevance networks and lead to less number of relevance networks.

The left-hand side of Fig. 5.5 shows that when TMI drops from 0.4 to 0.14, the number of relevance networks increases. Likewise, the number of nodes increases as illustrated in the right hand side of the figure. More decrease in TMI beyond 0.14 results in less number of relevance networks and a higher number of nodes. That means that new introduced edges lead to merge relevance networks together. The TMI is then fixed at the value that leads to the highest number of relevance networks, which is in this example 0.14.

After selecting the threshold, feature groups in the source domain D_{src} can be constructed from each relevance network. Afterwards, for each f_{indep} in each document of D_{trg}, the feature group is specified, and then the feature is linked to a sentiment with the highest mutual dependency within the group. An overall sentiment is then assigned to the document. The results are reported in Table 5.3.

Table 5.3 shows the MI model performance when relevance networks are used to assess the sentiment. The table shows improvement in the classification accuracies from the baseline model for some of the cases, while in some others the accuracies have dropped. In all of the cases, the lack of accuracies is due to the wrong classification of negative documents. Table 5.3 demonstrates very high precision values in almost all the cases, reaching above 95% in many cases. That indicates

Source domain	Target domain	Accuracy	Precision	Recall	F.measure	Neg Rate
music	computers	57.14	92	54.76	68.65	59.52
books	computers	57.14	84	55.26	66.66	59.02
movies	computers	75.51	88	70.96	78.57	80.06
hotel	computers	63.26	60	65.21	62.50	61.31
music	books	51.00	48	52.17	50.00	49.83
computers	books	51.02	96	51.06	66.66	50.98
movies	books	61.22	68	60.71	64.15	61.73
hotel	books	67.34	88	62.85	73.33	71.83
music	movies	75.50	84	72.41	77.77	78.59
computers	movies	61.22	96	57.14	71.64	65.30
books	movies	59.18	84	56.75	67.74	61.61
hotel	movies	53.06	84	52.50	64.61	53.62
music	hotels	65.30	100	59.52	74.62	71.08
computers	hotels	51.02	96	51.06	66.66	50.98
books	hotels	61.22	88	57.89	69.84	64.55
movies	hotels	65.30	96	60.00	73.84	70.60
computers	music	63.26	100	58.13	73.52	68.39
books	music	59.18	92	56.09	69.69	62.27
movies	music	81.63	96	75.00	84.21	88.26
hotel	music	51.00	60	51.72	55.55	50.28

Table 5.3: The evaluation metric for the MI model of cross-domain sentiment analysis. The averages are 61.52, 85, 59.05, 69, and 63.99, for accuracy, precision, recall, F-measure, and negative rate respectively. The highest levels of accuracies are achieved when movie is the source domain with an average of 70.91%

that positively predicted documents have a high probability to be true positive with an average of 85%. The average accuracy for positive documents (recall column) is 59.05%, while the average accuracy for negative documents (neg rate column) is 63.99%. Wilcoxon signed rank test is used to test whether there is a significant difference between the recall and true negative rate. The null hypothesis (H0) of the test is there is no significant difference between the two values. The alternative hypothesis (H1) is there is a significant difference between the two values. The test returned a p-value of 0.1595 which indicates that the null hypothesis cannot be rejected, and there is no significant difference in the prediction power of positive and negative documents using the MI model.

The wide range of accuracies can be due to two reasons in addition to domain dependency. First, for each experiment the features are selected from a different source domain from other experiments. This selection dominates the independent features selection in the target domain. The higher the number of common features is between the source and target domain the higher is the possibility for the classifier to find similarities between the two documents and infer sentiment. Second, the sizes of feature sets are small due to the small size of the data, and that is why some accuracies are close to random choice selection.

5.4.3 Infer Sentiment from Similarities

In this model, texts are represented as vectors. Each element in the vector of one document expresses the number of occurrences of the respective word in the document.

This model infers the sentiment between different domains using the similarity measure: the cosine. The similarity is quantified in this sense by computing the cosine between the documents, represented by vectors in the target domain D_{trg} and the source domain D_{src}. If a, and b are documents vectors, the cosine is defined by:

$$a \cdot b = \|a\| \cdot \|b\| \cdot \cos\theta \tag{5.4.3}$$

where θ is the angle between the two vectors. The range of cosine value is $[-1, 1]$, where -1 is the absolute dissimilarity, and 1 is the absolute similarity. However, in this analysis, the value of cosine is restricted to $[0, 1]$, as features are expressed with their frequencies in a text, and those frequencies cannot be negative. If all values are positive, then the maximal dissimilarity is orthogonal, and $a.b = 0$, then $\cos\theta = 0$. If a and b are co-directional, then $a \cdot b = \|a\| \cdot \|b\|$, and $\cos\theta = 1$, and this is the case of absolute similarities.

Knowing the source domain labels, each document in the target domain is assigned a label based on its similarity to the documents in source domain. To compute the cosine between two documents, two different representations of features in a document are used: numeric, and semantic representations. They are explained in the following.

NUMERIC REPRESENTATION MODEL (NRM)

Numeric feature representation is the simplest approach of text representation where each document is represented by a word/feature vector that reflects the features frequencies values. It is the same as constructing the $F^{(b)}$ matrices for each domain with an FF weighting as in Chapter 3. Both domains are represented using only the previously chosen independent features f_{indep} for D_{trg}. The cosine value is then computed pairwise for each document from D_{trg} with all documents from D_{src}. Having computed similarities, they are averaged over both and compared categories for each document in the target domain, then a sentiment score is assigned for the target domain with the highest similarity to the categories. The results are shown in Table 5.4.

In Table 5.4, a range of accuracies among the five different domains is presented, with an average of 64.6%, and a highest accuracy achieved 72% from movie to books domains. The accuracies have improved in comparison to the MI model, while the precision has dropped from an average of 85% in the MI model to an average of 61.8% in this model. The average accuracy for negative documents (true negative rate) is 61.82%, while the average accuracy for positive documents is 67.37%. Wilcoxon signed rank test of the accuracies of positive and negative predictions returns a p-value of 0.02361. Therefore, the difference in the positive and negative prediction is significant, and the NRM has a better predictability of positive documents than in negative documents, with an average difference of 5.55%.

Source domain	Target domain	Accuracy	Precision	Recall	F.measure	Neg Rate
music	computers	60	88	56.41	68.75	63.59
books	computers	64	44	73.33	55.00	54.67
movies	computers	64	44	73.33	55.00	54.67
hotel	computers	68	56	73.68	63.63	62.32
music	books	58	48	60.00	53.33	56.00
computers	books	70	56	77.78	65.11	62.22
movies	books	72	92	65.71	76.66	78.29
hotel	books	70	56	77.77	65.11	62.23
music	movies	64	44	73.33	55.00	54.67
computers	movies	64	44	73.33	55.00	54.67
books	movies	56	52	56.52	54.16	55.48
hotel	movies	64	44	73.33	55.00	54.67
music	hotels	68	76	65.51	70.37	70.49
computers	hotels	64	72	62.06	66.66	65.94
books	hotels	70	64	72.72	68.08	67.28
movies	hotels	70	64	72.72	68.08	67.28
computers	music	60	88	56.41	68.75	63.59
books	music	62	64	61.53	62.74	62.47
movies	music	62	56	63.63	59.57	60.37
hotel	music	62	84	58.33	68.85	65.67

Table 5.4: The evaluation metric of the NRM for cross-domain sentiment analysis with numeric representation. The model shows a more balanced outcome in terms of negative and positive prediction. The averages are 64.6, 61.8, 67.3714, 62.74, and 61.82 for accuracy, precision, recall, F-measure, and negative rate respectively.

Semantic Representation Model (SRM)

Unlike the numeric representation that uses features syntactics, semantic features representation relies on the features' context to determine their relatedness to other features in the texts. The method that is used for this representation is called latent semantic analysis (LSA). LSA applies statistical techniques to a corpus to represent the "contextual-usage meaning" of features [64]. LSA is usually applied on feature frequency matrices without-pre-processing of the source domain D_{src}, which is the $F^{(b)}$ matrix with FF weight in this thesis.

Before applying LSA, the matrix is subject to some transformations[3] to estimate the features importance. For each feature in each document, the log frequency is

[3] different from transformation in Chapter 3

Source domain	Target domain	Accuracy	Precision	Recall	F-measure	Neg Rate
music	computers	58	80	55.55	65.57	60.45
books	computers	68	60	71.42	65.21	64.58
movies	computers	56	52	56.52	54.16	55.48
hotel	computers	52	56	51.85	53.84	52.15
music	books	56	80	54.05	64.51	57.95
computers	books	64	64	64.00	64.00	64.00
movies	books	64	68	62.96	65.38	65.04
hotel	books	50	52	50.00	50.98	50.00
music	movies	66	80	62.50	70.17	69.50
computers	movies	58	56	58.33	57.14	57.67
books	movies	70	68	70.83	69.38	69.17
hotel	movies	52	52	52.00	52.00	52.00
music	hotels	68	72	66.66	69.23	69.34
computers	hotels	60	72	58.06	64.28	61.94
books	hotels	60	80	57.14	66.67	62.86
movies	hotels	58	60	57.69	58.82	58.31
computers	music	56	56	56.00	56.83	56.00
books	music	72	84	67.74	75.00	76.26
movies	music	60	56	60.86	58.33	59.14
hotel	music	56	60	55.55	57.69	56.45

Table 5.5: The evaluation metric of the SRM for cross-domain sentiment analysis with semantic representation using LSA method. The averages are 60.2, 65.4, 59.48, 61.95, and 60.91 for accuracy, precision, recall, F-measure, and negative rate respectively.

computed, and then the feature entropy divided by the entropy of the document it appears in is calculated. After that, LSA utilises this matrix to create LSA space distance matrix in which the similarities computations will be conducted on. More details about LSA can be found in [64]. The similarity is then assessed between documents in D_{src} and documents in D_{trg}, using the LSA space of D_{src} for computations. After that the sentiment scores are assigned to the documents in D_{tgr} based on the highest average similarity score measured on both categories. The results are presented in Table 5.5

As Table 5.5 shows, the similarity model using LSA representation has not shown an improvement of the results from the NRM. Also, some accuracies have dropped and only few has improved. The average accuracy has dropped to 60.2% as well as in the rest of the measures. Furthermore, the predictability for both negative and positive documents is the same. The following is a thorough comparison between the previous presented models for cross domain analysis.

5.4.4 Models Comparison

In the following, the baseline prediction is denoted by “BL”. The MI model is denoted by “MI”. The similarity model using numerical representation is denoted by “NRM”, and the similarity model using LSA is denoted by “SRM”. Fig. 5.6 compares the prediction accuracies/power of each source domain to the other four target domains using the four previously introduced models. The figure shows that the highest accuracy is achieved when the source domain is movies using MI, that is, 70.92%. For the rest of the domains, MI shows similar accuracies to those achieved using the BL model. NRM shows a slight improvement in the accuracies for domains: music, books, hotels and computers. Apart from the drop in the hotel domain, SRM shows similar predictability for the rest of the domains.

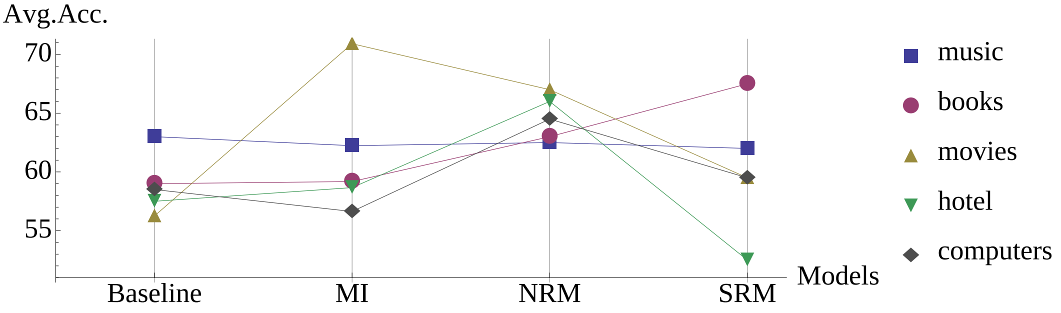


Figure 5.6: The average accuracies of each source domain prediction of the target domains for all models

To test whether the average accuracies in the four models differ significantly from each other, Wilcoxon signed rank test is applied for each two models. The null hypothesis says that the accuracies do not differ significantly. The results are shown in Table 5.6, which indicates a significant difference between NRM and all the other models. The rest of the comparisons show no significant difference in the prediction power from each other apart from MI that performed better than SRM. Thus, NRM performs best among all other models followed by MI.

model 1	model 2	p-value
BL	MI	0.29
BL	NRM	0.001
BL	SRM	0.46
MI	NRM	0.03
MI	SRM	0.3
NRM	SRM	0.009

Table 5.6: Wilcoxon signed rank test results, computed on accuracies of the models: BL, MI, NRM, SRM

To test the significance of the predictability of each source domain and each method, Wilcoxon signed rank test is computed for the accuracies achieved by each source domain. The null hypothesis says that the accuracies achieved by one source domain do not differ significantly from those achieved by another source domain.

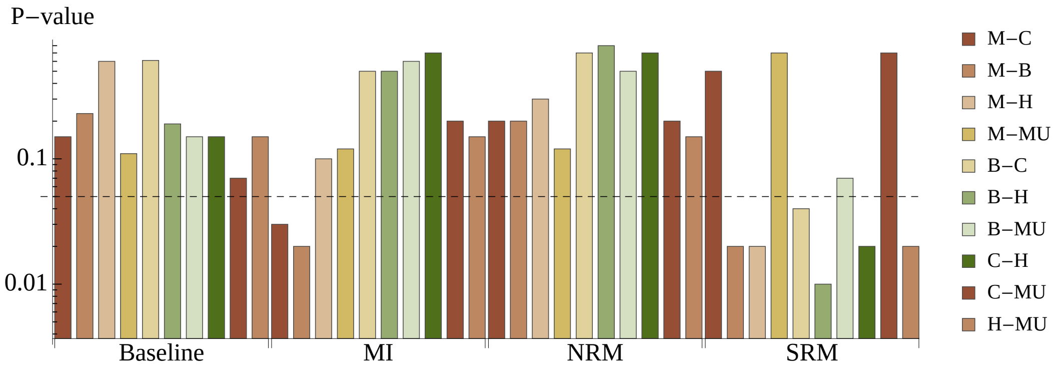


Figure 5.7: A bar chart representing the different p-values results of Wilcoxon signed rank test for each source domain to all target domains, in each model. Domains computers, movies, books, hotels and music are denoted by C, M, B, H, and MU respectively. Each bar represents the p-value of Wilcoxon test for the accuracies of two source domains

Fig. 5.7 demonstrates whether the accuracies achieved when using one source domain is significantly different from those accuracies achieved by using another source domain. For MI, the achieved prediction accuracies when using movies as a source domain are significantly better than when using computers or books as source domains. The accuracies in NRM for all source domains have no significant difference from each other. For SRM, using books as a source domain is significantly better in terms of prediction accuracies than using any other domain.

Fig. 5.8 illustrates the precision values of the prediction of each model. The MI model achieves the highest precision among other models with an average of 85%. That means that the probability is high that a document from the set of positively predicted documents is a true positive document.

With a null hypothesis that precision values do not differ significantly between different models, the Wilcoxon signed rank test shows that the average precision using MI is significantly higher than other models with a p-value less than 0.0001 for all other domains.

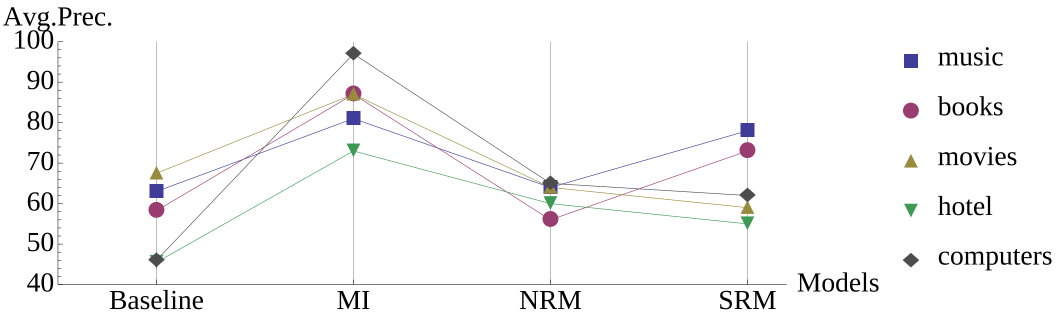


Figure 5.8: Average precision values for each domain in each model. The averages for all domains in each model are 56.05, 85, 61.8, and 65.4 for BL, MI, NRM, and SRM respectively

Fig. 5.9 compares the true negative rate for all models. The figure demonstrates relatively close negative predictability for all models. However, the MI model has a better negative predictability when the domain music is used as a source domain. According to Wilcoxon test, with a null hypothesis that the models true negative rates do not differ significantly from each other, the MI performs significantly better than BL and SRM in predicting negative documents, while it reaches the performance of NRM. Likewise, NRM performs significantly better than BL and SRM and reaches the performance of MI.

Finally, the models are compared from the target domain perspective; the target domain predictability averaging over all source domains for each model is tested.

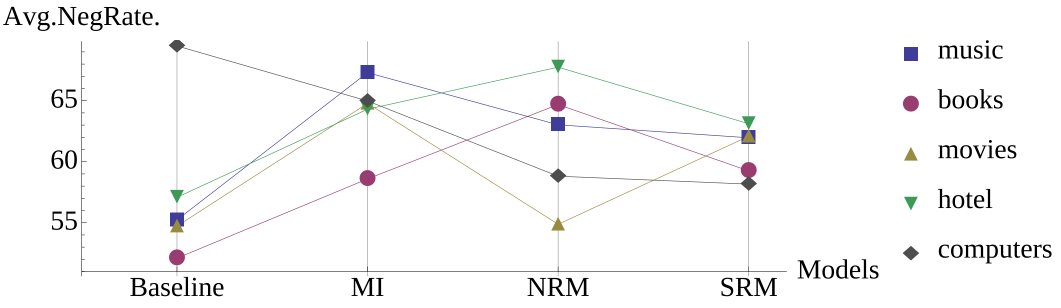


Figure 5.9: Average true negative rate values for each domain in each model. The averages for all domains in each model are 57.72, 63.99, 61.82, 60.91 for BL, MI, NRM, and SRM respectively.

Fig. 5.10 shows that the computer domain is best predicted using the BL model with an average accuracy of 70%, with significant difference from the performance of MI and NRM with average accuracies 64% and 63.26% respectively. The books domain is best predicted using NRM with an average accuracy of 67.5%. This average accuracy is significantly higher than in any other model with a Wilcoxon test p-values all smaller than 0.03. Likewise, the hotels domain is best predicted by NRM with an average accuracy of 68%. This accuracy is significantly higher than in other models with a Wilcoxon test p-values all smaller than 0.02. The music domain is best predicted by MI and NRM with average accuracies of 62.24% and 62% respectively. Finally, music domain is best predicted by MI and NRM with average accuracies of 63.76 and 61.5 respectively.

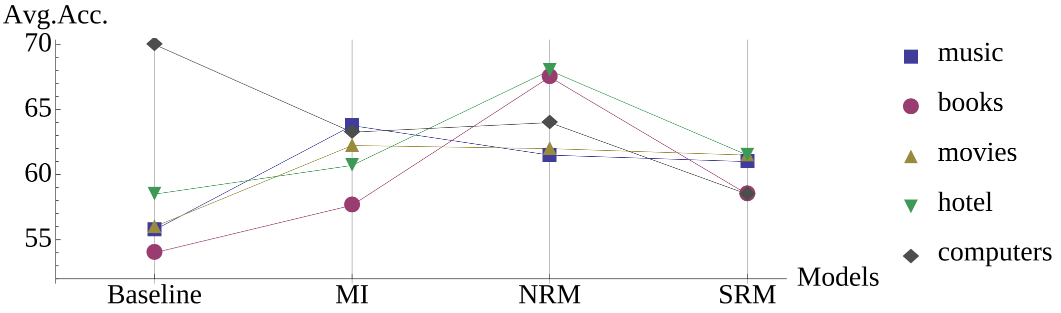


Figure 5.10: Average accuracies for all domains from the target domain perspective.

To conclude, the cosine similarity model using numerical representation (NRM), has the best performance among all analysed models for cross domain sentiment analysis followed by the mutual information model (MI).

5.5 UNSUPERVISED LEXICAL CLASSIFIER

An unsupervised lexical classifier can be built to be applied on all domains since it neither needs training, nor requires labelled data. In this section, an attempt to build a lexical classifier is explained, and evaluated. The algorithm is used on Dat-1400. First, the data is pre-processed, and then the algorithm is applied.

5.5.1 DATA PRE-PROCESSING

For this algorithm, the data is pre-processed differently. First, transformation is applied on the data and $F^{(t)}$ is constructed. Since it is assumed that the data is not labelled, χ^2 cannot be used as a filtering method, because it relies on the labels to measure the features significance. Therefore, a different method is used for filtering. For each document in Dat-1400, the document is split into sentences. Then for each sentence, a part of speech tagger (POS) is used on the data to identify the syntactic position for each feature. For example, " The actors choices are great", will be tagged as "The_DT actors_NNS choices_NNS are_VBP great_JJ", where DT refers to determiners, NNS refers to plural nouns, VBP refers to verbs or non-3rd person singular present, and JJ refers to adjectives. The pre-processing produces then for each document a list of sentences that contain features with their tags.

The algorithm utilises the hypothesis that negative and positive statements are mostly expressed by adjectives, adverbs and verbs [83, 31]. Therefore, the selected tags are: JJ (adjective), JJR (adjective, comparatives), JJS (adjective, superlative), RB (adverb), RBR (adverb, comparative), RBS (adverb, superlative), VB (Verb, base form), VBD (verb, past tense), VBG (verb, gerund or present participle), VBN (verb, past participle), VBP (verb, non 3rd person singular present), and VBZ (verb, 3rd person singular present).

The lexicon that is employed to identify the words polarities scores is SentiWordNet 3.0. It is constructed by assigning values to all the synsets (groups of one or more synonyms) of WordNet 3.0 according to their positivity, negativity or neutrality. It consists of more than 117000 synsets. SentiWordNet was explicitly created to be used for sentiment classification purposes [8].

5.5.2 ALGORITHM

After pre-processing, the classifier is used to do the following.

1. Divide each document into sentences.

2. For each sentence in each document, detect the features with the following tags: JJ, JJR, JJS, RB, RBR, RBS, VB, VBD, VBG, VBN, VBP, VBZ.

3. For each selected feature in each sentence compute the TF-IDF score.

4. For each selected feature in each sentence, use SentiWordNet 3.0 to pick up the feature's sentiment score.

5. Multiply the sentiment score with the TF-IDF score.

6. Aggregate the scores over sentences and then over documents using weighted averages.

7. If the aggregated score is bigger than zero then the document is assigned a positive label. On the contrary, if the aggregated score is smaller than zero then the document is assigned a negative label.

The prediction results are presented in Table 5.7.

Model	Accuracy	Precision	Recall	F-measure
Lexical	60	63	60	61
BL	58.85	56.05	59.97	56.63
MI	61.52	85	59.05	69
NRM	64.6	61.8	67.37	62.74
SRM	60.2	56.4	59.48	61.95

Table 5.7: Lexical classifier's evaluation metric in comparison to the previous models

Table 5.7 shows that the lexical classifier outperforms the baseline classifier introduced earlier, and performs as well as SRM and MI. However, the lexical classifier performance is significantly lower than the performance of NRM. That affirms the conclusion of the previous section that NRM outperforms other models that are proposed for cross-domain sentiment analysis in this chapter.

5.6 CONCLUSION

In this chapter, the problem of cross domain sentiment analysis has been investigated. The main focus of this chapter is the extraction of domain independent features from both source and target domain, to build a model to infer the sentiment of unlabelled data. First, the selection of domain independent features is inspected. A new method that defines a threshold to select those features is proposed. The outcome of this method is compared to the outcome of mutual information. Then, both methods are combined to assure the best selection of domain independent features. After selecting domain independent features, they are used in various approaches of cross-domain sentiment classification. Five domains are used in this chapter: movies, music, computers, books and hotels.

First, a baseline model (BL), utilising the sentiment classification algorithm from Chapter 3, is used to train each domain and predict the other domains. The performance is weak, with average accuracy of 58.85%. Then, a new model that is based on mutual information and relevance networks (MI) is introduced. The model achieves better accuracies than BL and high precision values, with an average accuracies of 61.52%, and average precision of 85%. Afterwards, two similarity-based models based on two different textual representations are introduced: numerical representation (NRM), and semantic representation (SRM). The average accuracies are 64.6% and 60.2% respectively. In an extensive analysis to compare the four models, NRM is identified as the best model among the others, followed by MI.

Finally a lexical model is applied on all domains. The great advantage of the lexical model is that it is, in contrast to all other models, an unsupervised model and therefore does not need any labelled data. The lexical model outperforms BL, and approaches SRM and then MI, while it is significantly lower than NRM.

CHAPTER 6

CONCLUSION AND FUTURE WORK

Sentiment analysis emerges as a challenging field with lots of obstacles as it involves natural language processing. It has a wide variety of applications that could benefit from its results, such as news analytics, marketing, question answering, knowledge bases and so on. The challenge of this field is to develop the machine's ability to understand texts as human readers do. Getting important insights from opinions expressed on the internet especially from social media blogs is vital for many companies and institutions, whether it is in terms of product feedback, public mood, or investors' opinions.

6.1 CONTRIBUTIONS

In this thesis, different aspects of sentiment analysis are under inspection. Text pre-processing techniques have been addressed widely in the information retrieval field. In particular the effect of various text pre-processing techniques is examined in the sentiment analysis field. Chapter 3 addresses the importance of pre-processing in boosting the sentiment classification performance. The sentiment of online movie reviews is investigated. A combination of different pre-processing methods are employed to reduce the noise in the text in addition to using the χ^2 method to remove irrelevant features that do not affect the text's polarity. Extensive experimental results have been reported, showing that, appropriate text pre-processing methods including data transformation and filtering can significantly enhance the classifier's performance. The level of accuracy that is achieved on the two data sets is comparable to the sort of accuracy that can be achieved in topic categorisation, a much easier problem.

Furthermore, the relationship between the number of features in a text and the prediction outcome of the classifier is investigated. It turned out the in the positive category the longer the document is, the higher are the probability and the confidence that it is classified correctly. As for the negative category, the probability of correct prediction is high even for small sized document. Furthermore, the impact of negative words is high on the classifier and that it is easier to predict negative classes for a lower number of features.

Chapter 3 illustrates the sentiment analysis problem from the opinion holder point of view. To have a thorough understanding of text sentiment, the reader point of view should be addressed, which has not been addressed in the literature to the best of my knowledge. Therefore, Chapter 4 delves into the reader sentiment and uses financial data as it allows for the capture of this type of sentiment. Two novel approaches toward financial news sentiment are explored: event-based and reader-based sentiment analysis. Event-based sentiment analysis detects the sentiment in the news according to the events that are announced in them. In order to achieve that, the relationship between stated events and the stock returns of the corresponding company have been examined. To comprehend this relationship, different parametric and nonparametric statistical methods have been employed. A simple Naive Bayes approach shows a strong relationship between event "alliance" and the returns. Further tests were then carried out to measure whether specific events has a statistical significant effect on the stock price. Those tests confirm the outcome of Naive Bayes approach: "alliance" event shows a significant effect on the returns.

A regression model is used to inspect the extent of this effect. After reducing the noise that originates from the market's activities, by using abnormal returns, the significance of the event "alliance" effect could not be re-established in a linear model.

Therefore, the proposed event-based model cannot be built in this case because no relationship could be found between almost all of the events and stock market returns. This conclusion can be data specific. Thus, the possible success of the event-based sentiment model cannot be ruled out. A bigger data set with higher frequencies of event occurrences could show a pattern that connects the events to stock market returns. Also, a non-linear model might explain the dynamics of this relationship. Another possible technique is to incorporate economic expert opinions about events that have an impact on stock markets in the model. To assess those options, more experiments on bigger data sets are to be carried out in the future.

As for the reader-based sentiment analysis, this thesis presents a novel model to detect the reader sentiment based on the stock market returns. It is argued in this thesis that the reader sentiment in financial news is reflected by the buy or sell decisions in the market, or in other words in the stock market returns. A new method to detect the reader sentiment and label the news accordingly is suggested (RSInd). To test the validity of this approach, SVM is used to classify the sentiment of the news. Afterwards, the predicted documents are used to test a trading strategy that uses this model against random strategy. As a result, the trading strategy that is built based on this model significantly outperforms the random strategy. This suggests that the reader-based sentiment model is a valid model for labelling the news and predicting their sentiment.

Financial news covers a wide range of domains, and that is one of the main reasons why the accuracies in Chapter 4 are lower than when using the same model that is used in Chapter 3 for movie data. Sentiment analysis is domain specific, and that creates the need for general classifiers that can cover the differences in the domains. In Chapter 5, the problem of cross domain sentiment analysis is investigated. The main focus of this chapter is to extract domain independent features from both source and target domain, and build a model to infer the sentiment of unlabelled data. First, the selection of domain independent features is inspected. A new method that defines a threshold to pick up those features is proposed. The outcome of this method is compared to the outcome of mutual information. Then, both methods are combined to assure the finest selection of domain independent features. After selecting domain independent features, they are used in various approaches of cross-domain sentiment classification. Five domains are used in this chapter: movies, music, computers, books and hotels.

First, a baseline model (BL) utilising the sentiment classification algorithm from Chapter3 is used to train each domain and predict the other domains. This model returned slightly low accuracies. Then, a new model that is based on mutual information and relevance networks (MI) is introduced and tested against the baseline model. The model achieves better accuracies than the BL model. Afterwards, two similarity-based models based on two different textual representations: Numerical representation (NRM), and semantic representation (SRM) are introduced. In an extensive analysis to compare the four models, NRM turns out to be clearly the best model among the others, followed by MI.

Finally a lexical model is applied on all domains. The lexical model outperforms BL, and performs as well as SRM and MI, but does significantly worse than NRM.

6.2 FUTURE RESEARCH

Much research can be carried out in the future. In financial news sentiment analysis, the results were data dependent. There is no suggestion that the event-based model cannot be a reliable model for sentiment prediction. Thus, more data needs to be collected to cover a wider variety of announced events and also have higher event's frequencies. This should allow for a deeper investigation of the relationship between event's occurrences and stock market returns, and therefore enables the reliability of event-based sentiment model to be tested.

As for the SVM classifier, other kernel function could be used in the classification. One example is string kernels which is a function that operates on strings. It allows SVM to deal with strings without the need to transform the input values into vector space representation.

The computational complexity of the classification problems in this thesis possibly poses a burden to the applicability of the presented results. In order to improve upon the computation times, a careful analysis of the algorithmic complexities is necessary. At the bottom of each of the discussed algorithms lies a machine learning problem. The complexity assessment of machine learning is however not directly comparable to the complexity analysis of classical algorithms such as sorting or searching. There exists a body of literature that advances in this area, see for instance [5, 54, 127]. It is left for future work to consider those theories and investigate the trade-off between additional performance and increased computation time.

As concluded earlier, the financial news covers different domains. A possible way to improve the classification accuracies is to incorporate topic categorisation in the sentiment analysis process. Then cross domain sentiment analysis methods can be used to improve the classification.

Relevance networks can be investigated further. One possible way is to inspect the features groups that emerge from different TMIs, and compute the pairwise mutual information between features from the target domain and features in the different groups in the source domain. This might allow for a better inferred sentiment score from one domain to another.

Finally, the NRM model can be improved by looking at deeper structures of similarities, such as computing sentences or words similarities, and then aggregating the similarities scores to a final score to infer the sentiment between two documents.

BIBLIOGRAPHY

[1] Abbasi, A., Chen, H., and Salem, A. Sentiment analysis in multiple languages: Feature selection for opinion classification in web forum. *ACM Transaction on Information Systems*, 26(3), 2008.

[2] Abbasi, A., France, S., Zhang, Z., and Chen, H. Selecting attributes for sentiment classification using feature relation networks. *Knowledge and Data Engineering, IEEE Transactions on*, 23(3):447–462, 2011.

[3] Agarwal, B. and Mittal, N. Optimal feature selection for sentiment analysis. In *Computational Linguistics and Intelligent Text Processing*, pages 13–24. Springer, 2013.

[4] Allison, B. Sentiment detection using lexically-based classifiers. In *Text, Speech and Dialogue*, pages 21–28. Springer, 2008.

[5] Angluin, D. Computational learning theory: survey and selected bibliography. In *Proceedings of the twenty-fourth annual ACM symposium on Theory of computing*, pages 351–369. ACM, 1992.

[6] Antweiler, W. and Frank, M. Z. Is all that talk just noise? the information content of internet stock messege board. *Journal of Finance*, 59(3):1259–1295, 2004.

[7] Aue, A. and Gamon, M. Customizing sentiment classifiers to new domains: A case study. In *Proceedings of recent advances in natural language processing (RANLP)*, volume 1, pages 2–1. Citeseer, 2005.

[8] Baccianella, S., Esuli, A., and Sebastiani, F. Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In *LREC*, volume 10, pages 2200–2204. 2010.

[9] Barbosa, L. and Feng, J. Robust sentiment detection on twitter from biased and noisy data. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, pages 36–44. Association for Computational Linguistics, 2010.

[10] Berry, T. D. and Howe, K. M. Public information arrival. *The Journal of Finance*, 49(4):1331–1346, 1994.

[11] Blair-Goldensohn, S., Hannan, K., McDonald, R., Neylon, T., Reis, G. A., and Reynar, J. Building a sentiment summarizer for local service reviews. In *WWW Workshop on NLP in the Information Explosion Era*, page 14. 2008.

[12] Blitzer, J., Dredze, M., and Pereira, F. Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In *Annual Meeting-Association For Computational Linguistics*, volume 45, page 440. 2007.

[13] Blitzer, J., McDonald, R., and Pereira, F. Domain adaptation with structural correspondence learning. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pages 120–128. Association for Computational Linguistics, 2006.

[14] Blood, D. J. and Phillips, P. C. Recession headlines news, consumer sentiment, the state of the economy and presedential popularity: a time series analysis 1989-1993. *International Journal of Public Opinion Research*, 7(1), 1995.

[15] Bollegala, D., Weir, D., and Carroll, J. Using multiple sources to construct a sentiment sensitive thesaurus for cross-domain sentiment classification. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, volume 1, pages 132–141. 2011.

[16] Bollegala, D., Weir, D., and Carroll, J. Cross-domain sentiment classification using a sentiment sensitive thesaurus. 2012.

[17] Bollen, J., Mao, H., and Zeng, X.-J. Twitter mood predicts the stock market. *Journal of computational science*, 2010. Doi:10.1016/j.jocs.2010.12.007.

[18] Bollen, J., Pepe, A., and Mao, H. Modeling public mood and emotion: Twitter sentiment and socio-economic phenomena. 2011. ArXiv:0911.1583v1 [cs.CY], Conference on Weblogs and Social Media (ICWSM 2011), 17-21 July 2011.

[19] Brys, G., Hubert, M., and Struyf, A. A robustification of the jarque-bera test of normality. Physica-Verlag/Springer, 2004.

[20] Butte, A. and Kohane, I. Mutual information relevance networks: Functional genomic clustering using pairwise entropy measurements. In *Pacific Symposium Biocomputing*, volume 5, pages 415–426. 2000.

[21] Charniak, E. *Statistical language learning*. MIT press, 1996.

[22] Chen, J., Huang, H., Tian, S., and Qu, Y. Feature selection for text classification with naïve bayes. *Expert Systems with Applications*, 36(3):5432–5435, 2009.

[23] Chen, Y.-T. and Chen, M. C. Using chi-square statistics to measure similarities for text categorization. *Expert systems with applications*, 38(4):3085–3090, 2011.

[24] Chen, Z., Mukherjee, A., Liu, B., Hsu, M., Castellanos, M., and Ghosh, R. Exploiting domain knowledge in aspect extraction. In *EMNLP*, pages 1655–1667. 2013.

[25] Chesley, P., Vincent, B., Xu, L., and Srihari, R. K. Using verbs and adjectives to automatically classify blog sentiment. *Training*, 580(263):233, 2006.

[26] Councill, I. G., McDonald, R., and Velikovich, L. What's great and what's not: learning to classify the scope of negation for improved sentiment analysis. In *Proceedings of the workshop on negation and speculation in natural language processing*, pages 51–59. Association for Computational Linguistics, 2010.

[27] Culotta, A. Towards detecting influenza epidemics by analyzing twitter messages. Published in: SOMA'10 Proceedings of the First Workshop on Social Media Analytics, 2010.

[28] Das, A., Bandyopadhyay, S., and Gambäck, B. Sentiment analysis: what is the end user's requirement? In *Proceedings of the 2nd International Conference on Web Intelligence, Mining and Semantics*, page 35. ACM, 2012.

[29] Das, S. R. *News Analytics: Framework, Techniques and Metrics*, chapter 2. Wiley Finance, 2010. The Handbook of News Analytics in Finance.

[30] Das, S. R. and Chen, M. Y. Yahoo! for amazon: Sentiment parsing from small talk on the web. In *EFA 2001 Barcelona Meetings*. 2001.

[31] Das, S. R. and Chen, M. Y. Yahoo! for amazon: Sentiment extraction from small talk on the web. *Management Science*, 53(9):1375–1388, 2007.

[32] Dave, K., Lawrence, S., and Pennock, D. M. Mining the peanut gallery: Opinion extraction and semantic classification of product reviews. In *Proceedings of WWW*, page 519528. 2003.

[33] Dehkharghani, R., Mercan, H., Javeed, A., and Saygin, Y. Sentimental causal rule discovery from twitter. *Expert Systems with Applications*, 41(10):4950–4958, 2014.

[34] Dimitriadou, E., Hornik, K., Leisch, F., Meyer, D., Weingessel, A., and Meyer, M. Package e1071. *R Software package, avaliable at http://cran. rproject. org/web/-packages/e1071/index. html*, 2009.

[35] Ding, X., Liu, B., and Yu, P. A holistic lexicon-based approach to opinion mining. In *Proceedings of the international conference on Web search and web data mining*, pages 231–240. ACM, 2008.

[36] Eirinaki, M., Pisal, S., and Japinder, S. Feature-based opinion mining and ranking. *Journal of Computer and System Sciences*, 2011.

[37] Fama, E. F. Market efficiency, long-term returns, and behavioral finance. *Journal of Financial Economics 49*, 1998.

[38] Feinerer, I., Hornik, K., and Meyer, D. Text mining infrastructure in r. *Journal of Statistical Software*, 25(5):1–54, 2008.

[39] Forman, G. Bns feature scaling: an improved representation over tf-idf for svm text classification. In *Proceedings of the 17th ACM conference on Information and knowledge management*, pages 263–270. ACM, 2008.

[40] Forman, G., Scholz, M., and Rajaram, S. Feature shaping for linear svm classifiers. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 299–308. ACM, 2009.

[41] Gao, S. and Li, H. A cross-domain adaptation method for sentiment classification using probabilistic latent analysis. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, pages 1047–1052. ACM, 2011.

[42] Go, A., Bhayani, R., and Huang, L. Twitter sentiment classification using distant supervision. *CS224N Project Report, Stanford*, pages 1–12, 2009.

[43] Hafez, P. *How News Events Impact Market Sentiment*, chapter 5. Wiley, 2010.

[44] Hall, M. A. and Smith, L. A. Feature subset selection: a correlation based filter approach. 1997.

[45] He, Y., Lin, C., and Alani, H. Automatically extracting polarity-bearing topics for cross-domain sentiment classification. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 123–131. Association for Computational Linguistics, 2011.

[46] Hsu, C.-W., Chang, C.-C., Lin, C.-J., et al. A practical guide to support vector classification. 2003.

[47] Hu, M. and Liu, B. Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 168–177. ACM, 2004.

[48] Hu, M. and Liu, B. Mining opinion features in customer reviews. In *Proceedings of the National Conference on Artificial Intelligence*, pages 755–760. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2004.

[49] Isa, D., Lee, L. H., Kallimani, V., and Rajkumar, R. Text document preprocessing with the bayes formula for classification using the support vector machine. *Knowledge and Data Engineering, IEEE Transactions on*, 20(9):1264–1272, 2008.

[50] Jensen, M. C. Some anomalous evidence regarding market efficiency. *Journal of Financial Economics*, 6(2), 1978.

[51] Joachims, T. Text categorization with support vector machines: Learning with many relevant features. *Machine Learning: ECML-98*, pages 137–142, 1998.

[52] Joachims, T. A statistical learning model of text classification for support vector machines. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 128–136. ACM, 2001.

[53] Kaji, N. and Kitsuregawa, M. Building lexicon for sentiment analysis from massive collection of html documents. In *Proceedings of the joint conference on empirical methods in natural language processing and computational natural language learning (EMNLP-CoNLL)*, pages 1075–1083. 2007.

[54] Kearns, M. J. *The computational complexity of machine learning*. MIT press, 1990.

[55] Kennedy, A. and Inkpen, D. Sentiment classification of movie reviews using contextual valence shifters. *Computational Intelligence*, 22(2, Special Issue on Sentiment Analysis):110–125, 2006.

[56] Khadjeh Nassirtoussi, A., Aghabozorgi, S., Ying Wah, T., and Ngo, D. C. L. Text mining for market prediction: A systematic review. *Expert Systems with Applications*, 41(16):7653–7670, 2014.

[57] Kim, S.-M. and Hovy, E. Determining the sentiment of opinions. In *COL-ING*. Association for Computational Linguistics, 2004.

[58] Kim, S.-M. and Hovy, E. Automatic detection of opinion bearing words and sentences. In *Companion Volume to the Proceeding of the International Joint Conference on Natural Language Processing*. 2005.

[59] Kim, S.-M. and Hovy, E. Automatoc identification of pro and con reasons in online reviews. In *Proceedings of COLING/ACL Main Conference Poster Session*, pages 483–490. Association for Computational Linguistics, 2006.

[60] Klussmann, A. G. and Hautsch, N. When machines read the news: Using automated text analytics to quantify high frequency news-implied market reactions. *Journal of Empirical Financ*, 18:321–340, 2011.

[61] Kobayashi, N., Inui, K., Matsumoto, Y., Tateishi, K., and Fukushima, T. Collecting evaluation expressions for opinion extraction. In *Proceeding of the International Joint Conference on Natural Language Processing (IJCNLP)*. 2004.

[62] Koncz, P. and Paralic, J. An approach to feature selection for sentiment analysis. In *Intelligent Engineering Systems (INES), 2011 15th IEEE International Conference on*, pages 357–362. IEEE, 2011.

[63] Ku, L., Lee, L., Wu, T., and Chen, H. Major topic detection and its application to opinion summarization. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 627–628. ACM, 2005.

[64] Landauer, T. K., Foltz, P. W., and Laham, D. An introduction to latent semantic analysis. *Discourse processes*, 25(2-3):259–284, 1998.

[65] Lee, C. and Lee, G. Information gain and divergence-based feature selection for machine learning-based text categorization. *Information processing & management*, 42(1):155–165, 2006.

[66] Lee, Y.-J. Reduced support vector machines: A statistical theory. *IEEE Trans. Neural Netw*, 18(1):1–13, 2007.

[67] Leinweber, D. and Sicks, J. Relating news analytics to stock returns. CARISMA, London, 2010.

[68] Lerman, K., Blair-Goldensohn, S., and McDonald, R. Sentiment summarization: evaluating and learning user preferences. In *Proceedings of the 12th Conference of*

the European Chapter of the Association for Computational Linguistics, pages 514–522. Association for Computational Linguistics, 2009.

[69] Li, Y., Luo, C., and Chung, S. M. Text clustering with feature selection by using statistical data. *Knowledge and Data Engineering, IEEE Transactions on*, 20(5):641–652, 2008.

[70] Liu, B. *Sentiment Analysis and Subjectivity. Handbook of Natural Language Processing*. Chapman & Hall/CRC Machine Learning & Pattern Recognition, second edition, 2010.

[71] Liu, B. Sentiment analysis and opinion mining. *Synthesis Lectures on Human Language Technologies*, 5(1):1–167, 2012.

[72] Liu, H., Sun, J., Liu, L., and Zhang, H. Feature selection with dynamic mutual information. *Pattern Recognition*, 42(7):1330–1339, 2009.

[73] Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., and Potts, C. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 142–150. Association for Computational Linguistics, 2011.

[74] Makrehchi, M. and Kamel, M. Automatic extraction of domain-specific stopwords from labeled documents. *Advances in information retrieval*, pages 222–233, 2008.

[75] McDonald, R., Hannan, K., Neylon, T., Wells, M., and Reynar, J. Structured models for fine-to-coarse sentiment analysis. In *Annual Meeting-Association For Computational Linguistics*, volume 45, page 432. Citeseer, 2007.

[76] Meesad, P., NuiPian, V., and Boonrawd, P. A chi-square-test for word importance differentiation in text classification. *Proceedings of Computer Science and Information Technology*, 6:110–114, 2011.

[77] Melville, P., Gryc, W., and Lawrence, R. Sentiment analysis of blogs by combining lexical knowledge with text classification. In *Proceedings of the 15th ACM*

SIGKDD international conference on Knowledge discovery and data mining, pages 1275–1284. ACM, 2009.

[78] Michell, M. L. and Mulherin, J. The impact of public information on the stock market. *The Journal of Finance*, 49(3):923–950, 1994.

[79] Mitra, L. and Mitra, G. *Application of news analytics in finance: A review*, chapter 1. John Wiley & Sons, 2011. The Handbook of News Analytics in Finance.

[80] Montoyo, A., Martínez-Barco, P., and Balahur, A. Subjectivity and sentiment analysis: An overview of the current state of the area and envisaged developments. *Decision Support Systems*, 53(4):675–679, 2012.

[81] Moreo, A., Romero, M., Castro, J., and Zurita, J. M. Lexicon-based comments-oriented news sentiment analyzer system. *Expert Systems with Applications*, 39(10):9166–9180, 2012.

[82] Mukherjee, A. and Liu, B. Discovering user interactions in ideological discussions. 2013.

[83] Na, J.-C., Sui, H., Khoo, C., Chan, S., and Zhou, Y. Effectiveness of simple linguistic processing in automatic sentiment classification of product reviews. In *Conference of the International Society for Knowledge Organization (ISKO)*, pages 49–54. 2004.

[84] Narayanan, R., Liu, B., and Choudhary, A. Sentiment analysis of conditional sentences. In *Proceeding of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 180–189. 2009.

[85] O' Keefe, T. and Koprinska, I. Feature selection and weighting methods in sentiment analysis. *ADCS 2009*, page 67, 2009.

[86] Pan, S. J., Ni, X., Sun, J.-T., Yang, Q., and Chen, Z. Cross-domain sentiment classification via spectral feature alignment. In *Proceedings of the 19th international conference on World wide web*, pages 751–760. ACM, 2010.

[87] Pang, B. and Lee, L. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, page 271. Association for Computational Linguistics, 2004.

[88] Pang, B. and Lee, L. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the ACL*. 2005.

[89] Pang, B. and Lee, L. Opinion mining and sentiment analysis. Foundation and Trends in Information Retrieval, 2008.

[90] Pang, B., Lee, L., and Vaithyanathan, S. Thumbs up? sentiment classification using machine learning. pages 97–86. Association for Computational Linguistics, 2002. Conference on Empirical Methods in Natural Language processing EMNLP.

[91] Paul, M. J. and Dredze, M. You are what you tweet: Analyzing for public health. Human Language Technology Center of Excellence, 2011. Center of language and speech processing.

[92] Peng, H., Long, F., and Ding, C. Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 27(8):1226–1238, 2005.

[93] Ponomareva, N. and Thelwall, M. Biographies or blenders: Which resource is best for cross-domain sentiment analysis? In *Computational Linguistics and Intelligent Text Processing*, pages 488–499. Springer, 2012.

[94] Ponomareva, N. and Thelwall, M. Do neighbours help?: an exploration of graph-based algorithms for cross-domain sentiment classification. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 655–665. Association for Computational Linguistics, 2012.

[95] Popescu, A. and Etzioni, O. Extracting product features and opinions from reviews. In *Proceedings of the conference on Human Language Technology and Empirical*

Methods in Natural Language Processing, pages 339–346. Association for Computational Linguistics, 2005.

[96] Porter, M. et al. An algorithm for suffix stripping. 1980.

[97] Pozzi, F. A., Fersini, E., and Messina, E. Bayesian model averaging and model selection for polarity classification. In *Natural Language Processing and Information Systems*, pages 189–200. Springer, 2013.

[98] Pozzi, F. A., Maccagnola, D., Fersini, E., and Messina, E. Enhance user-level sentiment analysis on microblogs with approval relations. In *AI* IA 2013: Advances in Artificial Intelligence*, pages 133–144. Springer, 2013.

[99] Prabowo, R. and Thelwall, M. Sentiment analysis: A combined approach. *Journal of Informetrics*, 3(2):143–157, 2009.

[100] Riloff, E., Patwardhan, S., and Wiebe, J. Feature subsumption for opinion analysis. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pages 440–448. Association for Computational Linguistics, 2006.

[101] Ross, S. A., Westerfield, R. W., and Jaffe, J. *Corporate Finance*. McGraw-Hill, seventh edition edition, 2005. International Edition.

[102] Rushdi Saleh, M., Martín-Valdivia, M. T., Montejo-Ráez, A., and Ureña-López, L. Experiments with svm to classify opinions in different domains. *Expert Systems with Applications*, 38(12):14799–14804, 2011.

[103] Russell, S. and Norving, P. *Artificial Intelligence: A Modern Approach*. Prentice Hall Artificial Intelligence Series. Pearson Education Inc., second edidtion edition, 2003.

[104] Sakaki, T., Okazaki, M., and Matsuo, Y. Earthquake shakes twitter users: real-time event detection by social sensors. In *Proceedings of the 19th international conference on World wide web*, pages 851–860. ACM, 2010.

[105] Schölkopf, B., Platt, J., Shawe-Taylor, J., Smola, A., and Williamson, R. Estimating the support of a high-dimensional distribution. *Neural computation*, 13(7):1443–1471, 2001.

[106] Scholkopf, B., Sung, K., Burges, C., Girosi, F., Niyogi, P., Poggio, T., and Vapnik, V. Comparing support vector machines with gaussian kernels to radial basis function classifiers. *Signal Processing, IEEE Transactions on*, 45(11):2758–2765, 1997.

[107] Sebastiani, F. Machine learning in automated text categorization. *ACM computing surveys (CSUR)*, 34(1):1–47, 2002.

[108] Shamshurin, I. Extracting domain-specific opinion words for sentiment analysis. In *Advances in Computational Intelligence*, pages 58–68. Springer, 2013.

[109] Si, J., Mukherjee, A., Liu, B., Li, Q., Li, H., and Deng, X. Exploiting topic based twitter sentiment for stock prediction. In *ACL (2)*, pages 24–29. 2013.

[110] Taboada, M., Anthony, C., and Voll, K. Methods for creating semantic orientation dictionaries. In *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC), Genova, Italy*, pages 427–432. 2006.

[111] Taboada, M. and Grieve, J. Analyzing appraisal automatically. In *Proceedings of AAAI Spring Symposium on Exploring Attitude and Affect in Text (AAAI Technical Re# port SS# 04# 07), Stanford University, CA, pp. 158q161. AAAI Press*. 2004.

[112] Täckström, O. and McDonald, R. Semi-supervised latent variable models for sentence-level sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2*, pages 569–574. Association for Computational Linguistics, 2011.

[113] Tan, C., Lee, L., Tang, J., Jiang, L., Zhou, M., and Li, P. User-level sentiment analysis incorporating social networks. *Arxiv preprint arXiv:1109.6018*, 2011.

[114] Tan, L., Na, J., Theng, Y., and Chang, K. Sentence-level sentiment polarity classification using a linguistic approach. *Digital Libraries: For Cultural Heritage, Knowledge Dissemination, and Future Creation*, pages 77–87, 2011.

[115] Tan, L., Na, J., Theng, Y., and Chang, K. Phrase-level sentiment polarity classification using rule-based typed dependencies and additional complex phrases consideration. *Journal of Computer Science and Technology*, 27(3):650–666, 2012.

[116] Tan, S., Wu, G., Tang, H., and Cheng, X. A novel scheme for domain-transfer problem in the context of sentiment analysis. In *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pages 979–982. ACM, 2007.

[117] Tang, H., Tan, S., and Cheng, X. A survey on sentiment detection of reviews. *Expert Systems with Applications*, 36(7):10760–10773, 2009.

[118] Tetlock, P., SAAR-TSECHANSKY, M., and Macskassy, S. More than words: Quantifying language to measure firms' fundamentals. *The Journal of Finance*, 63(3):1437–1467, 2008.

[119] Thelwall, M., Buckley, K., and Paltoglou, G. Sentiment in twitter events. *Journal of the American Society for Information Science and Technology*, 62(2):406–418, 2011.

[120] Thelwall, M., Buckley, K., Paltoglou, G., Cai, D., and Kappas, A. Sentiment strength detection in short informal text. *Journal of the American Society for Information Science and Technology*, 61(12):2544–2558, 2010.

[121] Tsai, C. and Hsiao, Y. Combining multiple feature selection methods for stock prediction: Union, intersection, and multi-intersection approaches. *Decision Support Systems*, 50(1):258–269, 2010.

[122] Turney, P. D. Thumbs up or thumps down? semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 417–424. 2002.

[123] Uchyigit, G. and Clark, K. A new feature selection method for text classification. *International Journal of Pattern Recognition and Artificial Intelligence*, 21(2):423, 2007.

[124] Van Rijsbergen, C. Information retrieval. dept. of computer science, university of glasgow. 1979.

[125] Van Rijsbergen, C., Robertson, S., Porter, M., and of Cambridge, U. *New models in probabilistic information retrieval*. University of Cambridge, Computer Laboratory, 1980.

[126] Vapnik, V. *The nature of statistical learning theory*. springer, 1995.

[127] Vapnik, V. N. and Chervonenkis, A. Y. On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probability & Its Applications*, 16(2):264–280, 1971.

[128] Wang, J., Neskovic, P., and Cooper, L. N. Training data selection for support vector machines. In *ICNC 2005. LNCS*, pages 554–564. International Conference on Neural Computation, 2005.

[129] Wiebe, J., Wilson, T., and Bell, M. Identifying collocations for recognizing opinions. In *Proceeding of the ACL/EACL Workshop on Collocation: Computational Extraction, Analysis and Exploitation*. 2001.

[130] Wilcoxon, F. Individual comparison by ranking methods. *Biometric Bulletin*, 1(6):80–83, 1945.

[131] Wilson, T., Wiebe, J., and Hoffmann, P. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of the Human Language Technology Conference and the Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP)*, pages 347–354. 2005.

[132] Wu, Q., Tan, S., and Cheng, X. Graph ranking for sentiment transfer. In *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*, pages 317–320. Association for Computational Linguistics, 2009.

[133] Xia, R. and Zong, C. Exploring the use of word relation features for sentiment classification. In *Proceedings of the 23rd International Conference on Computational*

Linguistics: Posters, pages 1336–1344. Association for Computational Linguistics, 2010.

[134] Yang, H., Callan, J., and Si, L. Knowledge transfer and opinion detection in the trec 2006 blog track. In *TREC*. 2006.

[135] Yi, J., Nasukawa, T., Bunescu, R., and Niblack, W. Sentiment analyzer: Extracting sentiments about a given topic using natural language processing techniques. In *Data Mining, 2003. ICDM 2003. Third IEEE International Conference on*, pages 427–434. IEEE, 2003.

[136] Yi, J. and Niblack, W. Sentiment mining in webfountain. In *Proceeding of the 21st International Conference on Data Engineering*. IEEE Computer Society, 2005. 1084-4627/05.

[137] Yu, H. and Hatzivassiloglou, V. Towards answering opinion questions: separating facts from opinions and identifying the polarity of opinion sentences. In *Proceedings of the conference on Empirical methods in natural language processing*, pages 129–136. EMNLP-2003, 2003.

[138] Yu, L. and Liu, H. Feature selection for high-dimensional data: A fast correlation-based filter solution. In *ICML*, volume 3, pages 856–863. 2003.

[139] Zhang, R. and Tran, T. An information gain-based approach for recommending useful product reviews. *Knowledge and Information Systems*, 26(3):419–434, 2011.

Appendices

APPENDIX A

FINANCIAL DATABASE STRUCTURE

Id	Symbol	Url	Title	Crawl Date	Content	Calais Results	Date Of Publishing	Publisher
13	AAL.L	http://us.rd.yahoo....	Estimated Dividend Yield ...	2011-07-04	<!DOCTYPE html PUBLIC ...	{"doc":{"info":{"...	2011-07-01 21:48:24	bloomberg
14	AAL.L	http://us.rd.yahoo....	South African Stocks: Fir...	2011-06-26	<!DOCTYPE html PUBLIC ...	{"doc":{"info":{"...	2011-06-22 17:01:31	bloomberg
15	AAL.L	http://us.rd.yahoo....	South African Fuel Strike...	2011-07-20	<!DOCTYPE html PUBLIC ...	{"doc":{"info":{"...	2011-07-20 17:30:28	bloomberg
16	AAL.L	http://us.rd.yahoo....	South Africa Stocks Snap ...	2011-06-26	<!DOCTYPE html PUBLIC ...	{"doc":{"info":{"...	2011-06-21 16:52:46	bloomberg
17	AAL.L	http://us.rd.yahoo....	South African Stocks: BHP...	2011-07-15	<!DOCTYPE html PUBLIC ...	{"doc":{"info":{"...	2011-07-15 16:52:37	bloomberg
18	AAL.L	http://us.rd.yahoo....	South African Stocks: Ang...	2011-06-26	<!DOCTYPE html PUBLIC ...	{"doc":{"info":{"...	2011-06-20 17:22:05	bloomberg
19	AAL.L	http://us.rd.yahoo....	Stocks in South Africa Ga...	2011-06-26	<!DOCTYPE html PUBLIC ...	{"doc":{"info":{"...	2011-06-24 16:55:29	bloomberg
20	AAL.L	http://us.rd.yahoo....	Copper Supply Squeeze Com...	2011-07-01	<!DOCTYPE html PUBLIC ...	{"doc":{"info":{"...	2011-07-01 06:19:00	cnbc
21	AAL.L	http://us.rd.yahoo....	UPDATE 1-Minara cuts nic ...	2011-06-30	<!--[if !IE]> This has...	{"doc":{"info":{"...	2011-06-30 06:53:00	reuters
22	AAL.L	http://us.rd.yahoo....	Diamond prices bounce ba ...	2011-07-08	<!--[if !IE]> This has...	{"doc":{"info":{"...	2011-07-08 14:01:00	reuters
23	AAL.L	http://us.rd.yahoo....	UPDATE 1-Chile Collahuasi...	2011-07-07	<!--[if !IE]> This has...	{"doc":{"info":{"...	2011-07-06 11:51:00	reuters
24	AAL.L	http://us.rd.yahoo....	Sundance gets \$1.5 billi...	2011-07-18	<!--[if !IE]> This has...	{"doc":{"info":{"...	2011-07-18 17:55:00	reuters
25	AAL.L	http://us.rd.yahoo....	UPDATE 2-SAfrica fuel wor...	2011-07-21	<!--[if !IE]> This has...	{"doc":{"info":{"...	2011-07-21 16:29:00	reuters
26	AAL.L	http://us.rd.yahoo....	S.African unions to strik...	2011-07-21	<!--[if !IE]> This has...	{"doc":{"info":{"...	2011-07-21 13:54:00	reuters
27	AAL.L	http://us.rd.yahoo....	Chile's Escondida mine re...	2011-06-26	<!--[if !IE]> This has...	{"doc":{"info":{"...	2011-06-19 09:28:00	reuters
28	AAL.L	http://us.rd.yahoo....	Chile Collahuasi mine say...	2011-07-07	<!--[if !IE]> This has...	{"doc":{"info":{"...	2011-07-06 10:18:00	reuters
29	AAL.L	http://us.rd.yahoo....	WRAPUP 4-Grid outage hits...	2011-06-26	<!--[if !IE]> This has...	{"doc":{"info":{"...	2011-06-19 12:51:00	reuters
30	AAL.L	http://us.rd.yahoo....	UPDATE 1-SAfrica strikes ...	2011-07-20	<!--[if !IE]> This has...	{"doc":{"info":{"...	2011-07-20 07:01:00	reuters
31	AAL.L	http://us.rd.yahoo....	S.Africa's NUM says coal ...	2011-07-07	<!--[if !IE]> This has...	{"doc":{"info":{"...	2011-07-06 17:09:00	reuters
32	AAL.L	http://us.rd.yahoo....	UPDATE 1-South Africa coa...	2011-07-25	<!--[if !IE]> This has...	{"doc":{"info":{"...	2011-07-25 10:01:00	reuters
33	AAL.L	http://us.rd.yahoo....	S.Africa braces for massi...	2011-07-25	<!--[if !IE]> This has...	{"doc":{"info":{"...	2011-07-24 14:00:00	reuters
34	AAL.L	http://us.rd.yahoo....	Chile's giant mines recov...	2011-07-11	<!--[if !IE]> This has...	{"doc":{"info":{"...	2011-07-09 06:09:00	reuters
35	AAL.L	http://us.rd.yahoo....	S.Africa's unions threate...	2011-07-07	<!--[if !IE]> This has...	{"doc":{"info":{"...	2011-07-07 11:44:00	reuters
36	AAL.L	http://us.rd.yahoo....	S.Africa coal miners star...	2011-07-25	<!--[if !IE]> This has...	{"doc":{"info":{"...	2011-07-25 08:53:00	reuters
37	AAL.L	http://us.rd.yahoo....	TABLE-Australia's top car...	2011-07-08	<!--[if !IE]> This has...	{"doc":{"info":{"...	2011-07-08 09:01:00	reuters
38	AAL.L	http://us.rd.yahoo....	UPDATE 1-S.Africa's NUM s...	2011-07-07	<!--[if !IE]> This has...	{"doc":{"info":{"...	2011-07-06 17:42:00	reuters
39	AAL.L	http://us.rd.yahoo....	RPT-Coal miners say Austr...	2011-07-11	<!--[if !IE]> This has...	{"doc":{"info":{"...	2011-07-10 10:46:00	reuters

Table A.1: Example of the financial news database structure. The table shows the relative information extracted for each news item. Each news item has an id, the ticker symbol of the corresponding company, the Url of the web page in which the news was obtained from, the title of the news article, the date when the news was crawled, the content of the webpage, the results of OpenCalais processing, the date when the news was published, and the publisher. Any other relative information that could be extracted from the news item is added to the table, such as the event the news is announcing, and any other related company to the event.

APPENDIX B

EXAMPLE OF OPENCALAIS RESULTS

```
”language”:”english”,”messages”:[]}},”http://d.opencalais.com/dochash-1/
55677ecf-bf3c-339e-8bd6-22edf9457812/cat/1”:{”_typegroup”:”topics”,
”category”:”http://d.opencalais.com/cat/calais/businessfinance”,
”classifiername”:”calais”,”categoryname”:”business_finance”,”score”:1},
”http://d.opencalais.com/generichasher-1/51874c83-ba21-36b7-a69e
-8a7cb4214b51”:{”_typegroup”:”entities”,”_type”:”industryterm”,
”name”:”energyassets”,”_typereference”:
”http://s.opencalais.com/1/type/em/e/industryterm”,
”instances”:[{”detection”:”[also has a division which invests directly in
]energy assets[, said on friday the latest investment would take]”,
”prefix”:”also has a division which invests directly in ”,
”exact”:”energy assets”,”suffix”:”, said on friday the latest
 investment would take”,”offset”:442,”length”:13}],
”relevance”:0.391},”http://d.opencalais.com/
generichasher-1/14cad6c8-8550-327c-9770-63b3358c19c5”:
{”_typegroup”:”entities”,”_type”:”country”,”name”:”nigeria”,
”_typereference”:”http://s.opencalais.com/1/type/em/e/country”,
”instances”:[{”detection”:”[statement.  petrofac said working with seven
in ]nigeria[ in the last six months had helped it identify]”,
”prefix”:”statement. petrofac said working with seven in”,
”exact”:”nigeria”,”suffix”:” in the last six months had helped it
 identify”,”offset”:1066,”length”:7}],”relevance”:0.189,”resolutions”
:[{”id”:”http://d.opencalais.com/er/geo/country
/ralg-geo1/f174114a-d69e-0db6-6eff-6090be60b5e5”,”name”:”nigeria”a”,
”offset”:955,”length”:20}],”relevance”:0.189},”http://d.opencalais.com
/generichasher-1/9443e26e-d592-3102-a098-4040aaf496c7”
...................................................
```

Listing B.1: Example of OpenCalais Results. OpenCalais processes the given text and extracts some relevant information from the texts. The type of information varies between several categories: facts, entities and events. For each of those information, the most relevant information to the study is the type and the relevance. The type allows to identify events, and they will be used in the event-based model. The relevance demonstrates how relevant the piece of information is to the corresponding company, and that is important for picking the most relevant events for the study.