Sentiment Analysis in Financial News

Patrícia Alexandra Lopes Antunes

2015

Master Thesis in Data Analytics

Supervised by Professor Pavel Brazdil

Dedicated to my husband Rui.

Biography

Patrícia Antunes was born on February 23, 1984 in Porto, Portugal. In 2009 she graduated in Economics at University of Porto. She was also an exchange programme student at Federal University of Rio de Janeiro in 2008.

Since 2009, she is a Business Analyst in Finantech, a software house that develops software for investment banking.

Stock markets and sentiment analysis are some of her main interests, and they have inspired Patrícia's master thesis in Data Analytics at University of Porto.

Acknowledgments

I would like to thank my supervisor, Professor Pavel Brazdil, for his guidance, support, incredible patience and excellent advices. I feel extremely lucky for having him as my supervisor.

To my family, thanks for understanding when I was absent and for encouraging me when I was present. In particular, I would like to thank my parents, my grandparents, my sister and my brother-in-law, but also my husband's parents and sister. To them all, a huge thanks.

To my beautiful niece Núria, that was born a couple of weeks before I started my work on the master degree. Sorry for not playing with you as many times as you wanted. It will be different from now on.

And finally, a very special thanks to my husband Rui, for embracing this challenge with me. For sitting next to me in all the classes of this master course and for giving me love and support while he was also writing his master thesis.

To all of you, my deepest gratitude.

Abstract

With the growth of social media, millions of financial news flow every day through the Web. This makes monitoring and interpreting what is happening in the financial world an extremely difficult task. Moreover, as Liu and Zhang (2012) pointed out, it is also known that human analysis of text information is subject to considerable biases. Therefore, a system for automatic detection of sentiment is extremely useful. This was the main motivation behind this work. We have decided to develop a system that can analyse news in the financial domain.

As textual data can be very noisy, text pre-processing techniques were applied to the news articles (e.g. *stopwords removal*, *stemming*). Afterwards, the news were classified as *positive*, *negative* or *neutral*, and a series of studies were carried out to improve the classification results. The method exploited several publicly available lexicons – *Opinion Lexicon*, *OpinionFinder*, *SentiWordNet*, *AFINN* and *NRC*. Moreover, we have merged some of these lexicons to see if the results could be improved. Besides, a negation handling technique developed by Pang et al. (2002) was also applied. Additionally, some words mainly from the financial world were added to the available lexicons. We have thus obtained enriched lexicons.

All experiments were evaluated using usual performance evaluation measures (e.g. *Micro F1*). However, as sentiment classification can be seen as a problem of classifying ordinal data, an evaluation using cost-sensitive analysis was carried out. That is, different costs were applied to different types of error.

We have obtained several interesting results. We have identified combinations of

2-3 lexicons that led to the best results. Negation handling did not always result in marked improvement. Finally, we have shown that enriched lexicons led to marked improvements of performance.

Keywords: Sentiment Analysis, Financial News, Cost-sensitive Analysis

Resumo

Com o crescimento dos meios de comunicação social, milhões de notícias circulam pela Web todos os dias. Isto faz com que monitorizar e interpretar o que se está a passar no mundo financeiro seja uma tarefa extremamente difícil. Para além disso, como Liu e Zhang (2002) referiram, é também sabido que a análise humana da informação de um texto pode ser tendenciosa. Assim sendo, um sistema para deteção automática de sentimento em notícias financeiras é extremamente útil. Esta foi a principal motivação por trás deste trabalho. Decidimos desenvolver um sistema que pode analisar notícias do domínio financeiro.

Como dados de texto podem ter muito ruído, técnicas de pré-processamento de texto foram aplicadas às notícias (p.e. *remoção de stopwords*, *stemming*). Posteriormente, cada notícia foi classificada como *positiva*, *negativa* ou *neutra*, e uma série de estudos foram realizados para melhorar os resultados da classificação. O método explorou vários léxicos disponíveis publicamente – *Opinion Lexicon*, *OpinionFinder*, *SentiWordNet*, *AFINN* e *NRC*. Para além disso, unimos alguns destes léxicos para ver se os resultados podiam ser melhorados. Adicionalmente, foi aplicada uma técnica de tratamento da negação desenvolvida por Pang et al. (2002). Foram ainda adicionadas mais palavras, principalmente do mundo financeiro, aos léxicos disponíveis. Obtivemos assim léxicos enriquecidos.

Todas as experiências foram avaliadas usando medidas de avaliação de performance usuais (p.e. *Micro F1*). No entanto, como a classificação de sentimento pode ser vista como um problema de classificar dados ordinais, foi realizada uma avaliação que usa uma análise sensível a custos. Isto é, diferentes custos foram atribuídos a

diferentes tipos de erros.

Obtivemos vários resultados interessantes. Identificámos combinações de 2-3 léxicos que levaram a melhores resultados. O tratamento da negação nem sempre resultou em melhoria acentuada. Finalmente, mostrámos que léxicos enriquecidos levaram a melhorias acentuadas da performance.

Palavras-Chave: Análise de Sentimento, Notícias Financeiras, Análise de Custos

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Motivation and the Problem Studied

With the growth of social media, millions of financial news flow every day through the Web. This massive volume of news makes monitoring and interpreting what is happening in the financial world an impossible task. Moreover, as Liu and Zhang (2012) stated, it is also known that human analysis of textual information is subject to considerable biases. It is known that, people often pay greater attention to opinions that are consistent with their own preferences.

If someone wants to invest in the stock market, financial news are a very important part of his/her decision making. If an investor had to read all the available news, that can be an overwhelming task. Therefore, if a system automatically filters the news focusing on those that have positive or negative sentiment attached, and discarding the ones that are neutral, then the task of analysing financial news is simplified.

This motivated us to define our goal which consists of developing a system for automatic detection of sentiment in financial news.

Sentiment analysis is the process of detecting the sentiment of a text. It determines whether it is *positive*, *negative* or *neutral*. In this work several sentiment

analysis techniques were used to extract sentiment from financial news.

The analysis of the news articles was performed using a lexicon-based approach. Therefore, publicly available sentiment lexicons were employed (*Opinion Lexicon*, *OpinionFinder*, *SentiWordNet*, *AFINN* and *NRC*) and used in the news sentiment classification. These sentiment lexicons consist of lists of words with assigned *positive* or *negative* value that reflects its sentiment polarity.

The first work carried out had the objective to determine which of these lexicons is better for the classification of financial news. Although we have obtained quite good results, we have decided to improve them further. We have decided to merge some of the publicly available lexicons and verified that almost all the merges that were tested improved the classification results.

Additionally, we have decided to incorporate the treatment of negation developed by Pang et al. (2002). It inverts the polarity of all words that are between a negation word (e.g. *not*, *isn't*, *didn't*) and the next punctuation mark. However, the results were not as good as expected, as only some negligible improvements were obtained.

All experiments were evaluated using evaluation measures that are appropriate for classification (e.g. *Micro F1*). However, as we are dealing with ordinal data, we note that classifying a positive news article as *neutral* is not as bad as classifying it as *negative*. Therefore, we adopted a cost-sensitive analysis, where different costs were applied to different types of error.

The last study was inspired by the fact that accuracy of sentiment classification can be highly sensitive to the text domain. Therefore, all the lexicons were enriched with more words from the financial world (e.g. *dividend*, *takeover*, *subprime*). This led to very positive results of overall performance.

1.2 Thesis Structure

The overall thesis is structured as follows:

Chapter 2 presents an overview of sentiment analysis. It starts by defining the sentiment analysis problem. Then, it describes the key concepts and methods that have been described in the literature and their representative techniques.

Chapter 3 describes the methods that have been used in this thesis. The corpus and the pre-processing techniques are presented. It also discusses different approaches to sentiment classification together with different evaluation techniques that have been used.

Chapter 4 describes our case study. In this chapter we also present the results of sentiment analysis for a series of experiments that involve news.

Chapter 5 presents the main conclusions and also some limitations of this work. It also describes the future work that could be done to improve our results.

Chapter 2

Overview of Sentiment Analysis

Sentiment analysis, also known as *opinion mining*, refers to the application of natural language processing, computational linguistics and text analytics to identify and extract subjective information in source materials (Batrinca and Treleaven, 2014).

However, when classifying sentiment, the source materials do not need to be an opinionated text. Classifying a news article into good or bad has been considered a sentiment classification task in the literature (Koppel and Shtrimberg, 2006; Ruiz-Martínez et al., 2012; Schumaker et al., 2012; Ahmad et al., 2006).

A news article can be classified into *positive* or *negative* without being opinionated, for example, if the news article refers that a *"company's profit rose"*. This is clearly good news, but if the news is about a *"company's bankruptcy"* then it is bad news.

In this chapter we present an overview of sentiment analysis research, mentioning key concepts, features, different levels of analysis, sentiment lexicons generation techniques and cost-sensitive analysis.

2.1 Key Concepts and Features

2.1.1 Sentiment Polarity and Subjectivity Detection

Different authors have dealt with the problem of sentiment classification in different ways.

Sentiment classification can be formulated either as two separate classification problems or as a three-class classification problem (Liu, 2012).

When formulated as a two separate classification problems, the first problem is to determine if a piece of text (e.g. a document) is subjective or objective, that is, if it expresses an opinion or not. This type of problem is called *subjectivity classification* (Hatzivassiloglou and Wiebe, 2000; Wilson et al., 2004; Wiebe et al., 2004). The second classification problem is to classify the subjective sentences into *positive* or *negative*. This binary classification task of labelling a document as expressing either an overall positive or an overall negative opinion is called *(sentiment) polarity classification* (Pang and Lee, 2008).

In case when the problem is defined as a three-class classification problem, the piece of text is classified either as *positive* or *negative* or *neutral*. In the literature, the label *neutral* is sometimes used for the objective class (*lack of opinion*) or only as the sentiment that lies between positive and negative (Pang and Lee, 2008).

However, sometimes this type of classification (*positive*/*negative*/*neutral*) is not considered satisfactory, as more information may be needed. Therefore, some authors (Pang and Lee, 2005; Goldberg and Zhu, 2006) have used multi-point scales in their work (e.g. one to five points). This type of classification may be viewed as a multi-class text categorization problem, or also *ordinal classification* that was described further on.

2.1.2 Term Presence vs. Frequency

The term presence approach uses binary values and simply determines if the term occurs (value 1) or not (value 0). In the term frequency approach the values reflect the number of occurrences of a term.

Term frequencies have been widely used, but in some cases better performance has been obtained using the binary instead of frequency (Pang et al., 2002). The author showed that while a topic is more likely to be emphasized by frequent occurrences of certain keywords, overall sentiment may not be highlighted through repeated use of the same terms.

2.1.3 Parts of Speech Tagging

In a parts of speech representation, words are assigned a *part of speech tag*. The traditional English parts of speech are noun, verb, adjective, adverb, pronoun, preposition, conjunction, and interjection. Words that are assigned to the same part of speech generally display similar behaviour in terms of syntax.

Some researchers have treated words with different parts of speech tag differently (Santorini, 1990; Hatzivassiloglou and McKeown, 1997; Hatzivassiloglou and Wiebe, 2000; Turney, 2002). It was shown that some adjectives are important indicators of opinions, and that some nouns are also strong indicators of sentiment (e.g. the nouns *hate* and *love*).

2.1.4 Negation

Negation words represent an important concept in sentiment analysis, as they invert polarity. They are sentiment shifters. The phrase *"People should not invest in this company"* is very similar to *"People should invest in this company"*, but from a

sentiment point of view, they are complete opposites.

Pang et al. (2002) adapted the technique of Das and Chen (2001) and added the tag NOT to every word between a negation word (e.g. *not*, *isn't*, *didn't*) until the first punctuation mark is encountered. So, if a word is preceded by a negation word, it will not be considered and the prefix NOT_ will be added to each word until the end of sentence (e.g. People should not NOT_invest NOT_on NOT_this NOT_company).

2.2 Levels of Analysis

Sentiment analysis has been investigated mainly at three levels: document level (for document-based sentiment), sentence level (for sentence-based sentiment) or entity and aspect level (for aspect-based sentiment).

In the following we present a brief description of these different levels of analysis.

2.2.1 Document Level Sentiment Analysis

Document level analysis classifies an entire document as expressing positive or negative sentiment (Pang et al., 2002; Turney, 2002).

As Liu (2010) stated, document level sentiment classification assumes that the document expresses opinions on a single entity and the opinions are from a single opinion holder. If we have documents that evaluate or compare multiple entities, this level of analysis is no longer sufficient and a greater level of detail can be obtained by applying sentence-level sentiment classification.

2.2.2 Sentence Level Sentiment Analysis

Sentence-level sentiment classification gives a more detailed view than document-level sentiment analysis. Moreover, the same techniques of document-level analysis

can be applied to sentences.

This level of analysis assumes that the sentence expresses a single opinion from a single opinion holder (Liu, 2010). However, this is not always the case. As Liu (2012) pointed out, many complex sentences have different sentiments on different targets, such as, *"BCP is recovering after BES bankruptcy."*.

Other difficulties of sentiment classification on a sentence level stem from the fact that it cannot deal with opinions in comparative sentences (e.g. *"BPI is doing better than BCP."*), sentences formulated as questions (e.g. *"Is BPI doing better than BCP?"*), and sarcastic sentences (e.g. *"BCP is doing so well!"* which may mean the exact opposite of what is the apparent content.

Some work has been developed to overcome these difficulties. Jindal and Liu (2006) have studied the problem of identifying comparative sentences in text documents and Tsur et al. (2010) have presented a way to identify sarcastic sentences.

Despite the fact that document level or sentence level analyses represent a good approach in many cases, they may not reach the level of detail required. In such cases an *aspect level analysis* provides a good alternative. It is discussed in the next section.

2.2.3 Entity and Aspect Level Sentiment Analysis

The two previous approaches perform very well when the whole document or each sentence refers to a single entity. However, texts may refer to different entities that can have many aspects, and the opinion about each entity or each topic may be different.

Aspect level was earlier called *feature level* (feature-based opinion mining and summarization) (Hu and Liu, 2004) and has the goal of discovering sentiments relative to entities and/or their aspects. Feldman (2013) defines aspect-based sentiment

analysis as the research problem that focuses on the recognition of all sentiment expressions within a given document and the aspects to which they refer.

The traditional approach to aspect level sentiment analysis is to identify all aspects in a corpus of product reviews and extract all noun phrases and then keep just the frequent noun phrases (Hu and Liu, 2004).

2.3 Lexicon-based Approaches

A sentiment lexicon is a list of words assigned with a positive or negative score reflecting its sentiment polarity. Examples of positive words are: *good*, *beautiful*, *happy* and *nice*. Examples of negative words are *bad*, *ugly*, *unhappy*, *poor*, and *terrible*.

To acquire the sentiment lexicon, that is, the opinion word list, three main approaches have been used: manual approach, dictionary-based approach, and corpus-based approach. The three approaches are discussed next.

2.3.1 Elaboration of Sentiment Lexicon

Manual Approach

Lexicons for lexicon-based approaches can be created manually by hand-tagging the chosen words in a dictionary. Some researchers have chosen this approach in the past. Taboada et al. (2011) refers in his work that he decided to create a lexicon manually because of the lack of stability for automatically generated lexicons.

However, this task can be very time-consuming and is rarely used.

Dictionary-based Approach

As Liu and Zhang (2012) described, the strategy requires that we start by collecting

a small set of opinion words manually with known orientations, and then to grow this set by searching in the WordNet (Miller et al., 1990) or Thesaurus for their synonyms and antonyms. The newly found words are added to the seed list. The iterative process is continued and stops when no more new words have been found. This approach is used in Hu and Liu (2004) and Kim and Hovy (2004). After the process has been completed, manual inspection can be carried out to remove and/or correct errors.

The advantage of using dictionary-based approach is the easiness of how a large number of sentiment words can be found. However, as a down side, it cannot distinguish opinion words that have different meanings in different contexts. For example, if we are talking about profit, than the word *increase* is positive. However, if we are talking of debt, it is negative. The sentiment orientation of *increase* is context dependent.

As the dictionary-based approach cannot capture the specific peculiarities of a specific domain, the corpus-based approach can better deal with this problem.

Corpus-based Approach

The methods in the corpus-based approach rely on syntactic or co-occurrence patterns and also a seed list of opinion words to find other opinion words in a large corpus (Liu and Zhang, 2012).

The corpus-based approach tries to solve the problem that the same word can be positive in one context and negative in another.

A key development in this area was the work of Hatzivassiloglou and McKeown (1997) who introduced the concept of *sentiment consistency*. Their strategy used a list of seed opinion adjectives to set of linguistic constraints or conventions on connectives to identify other adjective sentiment words and their orientations from

the corpus.

One of the constraints involves the conjunction AND. It states that conjoined adjectives usually have the same orientation. For example, consider the sentence, "*This company is reliable and efficient*". If *reliable* is known to be positive, it can be inferred that *efficient* is also positive. This is so because people usually express the same opinion in both parts of a conjunction.

It can be noted that the following sentence is rather unnatural, "*This company is reliable and inefficient*". If it is changed to "*This company is reliable, but inefficient*", it becomes acceptable.

Rules or constraints were also designed for other connectives, OR, BUT, EITHER-OR, and NEITHER-NOR. This idea is what is referred to as *sentiment consistency*. However, in practice the terms are not always consistent. A learning step has been applied to a large corpus to determine whether two conjoined adjectives have the same or different orientations.

However, as Liu and Zhang (2012) stated, using the corpus-based approach alone to identify all opinion words is usually not as effective as the dictionary-based approach, because it is hard to prepare a huge corpus to cover all English words. Nevertheless, this approach is able to find some domain and context specific opinion words and their orientations using domain corpora.

2.3.2 Resources

As previously referred, a sentiment lexicon is a list of words accompanied with a positive or negative score reflecting its sentiment polarity and strength.

The development of lexicons for sentiment analysis has attracted the attention of the computational linguistic community. Various researchers have constructed

sentiment lexicons and some of them are publicly available. Some examples are:

- **ANEW** (Bradley and Lang, 1999)

  The lexicon ANEW (Affective Norms for English Words) provides a set of normative emotional ratings for a large number of words in the English language. The goal was to develop a set of textual materials that have been rated in terms of pleasure (ranging from pleasant to unpleasant), arousal (ranging from calm to excited), and dominance (or control) (Bradley and Lang, 1999).

  As ANEW was released before the rise of microblogging, it does not include many slang words. To overcome this disadvantage, improved versions of ANEW were developed later (e.g. AFFIN).

- **Opinion Lexicon** (Hu and Liu, 2004)

  Hu and Liu (2004) developed a lexicon that contains a sentiment list of about 6,800 words classified into positive or negative.

  It was generated using a bootstrapping strategy with some given positive and negative sentiment word seeds and the synonyms and antonyms relations in WordNet. It was compiled over many years starting from their first paper (Hu and Liu, 2004).

- **OpinionFinder** (Wilson et al., 2005a)

  The OpinionFinder Lexicon (OPF) is a polarity oriented lexical resource. It is an extension of the *Multi-Perspective Question-Answering* dataset (MPQA) (Wilson et al., 2005b).

  Each sentence was manually tagged according to the polarity classes: positive, negative, neutral. Then, a pruning phase was conducted over the dataset to

eliminate tags with low agreement.

- **SentiWordNet** (Esuli and Sebastiani, 2006)

  Esuli and Sebastiani (2006) extended the Wordnet (Miller et al., 1990) lexical database by introducing sentiment ratings to a number of synsets, creating SentiWordnet. Each WordNet synset s is associated to three numerical scores Obj(s), Pos(s), Neg(s), describing the degree of how *objective*, *positive*, or *negative* the terms contained in the synset are.

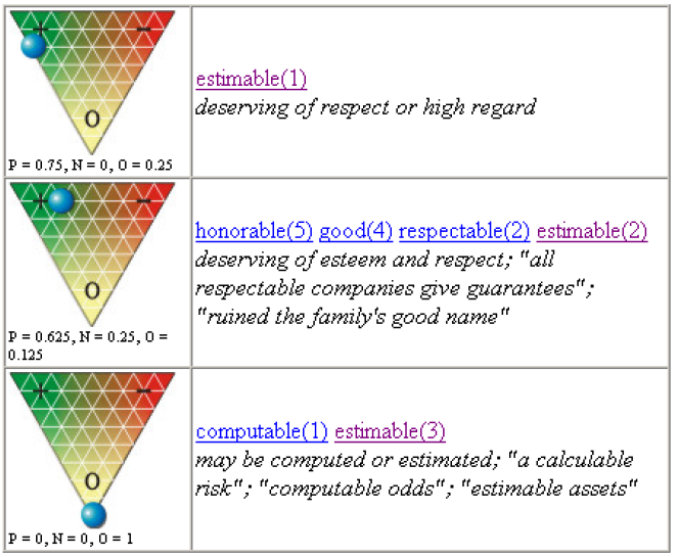

Figure 2.1: SentiWordNet visualization of the opinion related properties of the term *estimable* (Esuli and Sebastiani, 2006).

The assumption is that different senses of the same term may have different opinion-related properties. The scores range from 0.0 to 1.0, and their sum is 1.0 for each synset. As Esuli and Sebastiani (2006) explained, this means that a synset may have nonzero scores for all the three categories, which would indicate that the corresponding terms have, in the sense indicated by the

synset, each of the three opinion-related properties only to a certain degree.

An example with the synset *estimable*, that is an adjective with three senses, can be seen in Figure 2.1.

- **AFINN** (Nielsen, 2011)

  Inspired in ANEW, Nielsen (2011) created the AFINN lexicon, a lexicon more focused on the language used in microblogging which includes 2,477 English words.

  The word list includes slang, obscene words, acronyms and web jargon. Scoring ranges from -5 (very negative) to +5 (very positive), reason why this lexicon is useful for strength estimation.

- **NRC Lexicon** (Mohammad and Turney, 2013b)

  NRC is a word lexicon that contains more than 14,000 distinct English words.

  Words were manually annotated, through Amazon's Mechanical Turk service, according to the Plutchik's wheel of emotion. Eight emotions were considered during the creation of the lexicon, joy-trust, sadness-anger, surprise-fear, and anticipation-disgust, which constitute four opposing pairs. This emotion opposition is displayed in Figure 2.2 by the spatial opposition of these pairs.

  Additionally, NRC words are tagged according to polarity classes: positive and negative.

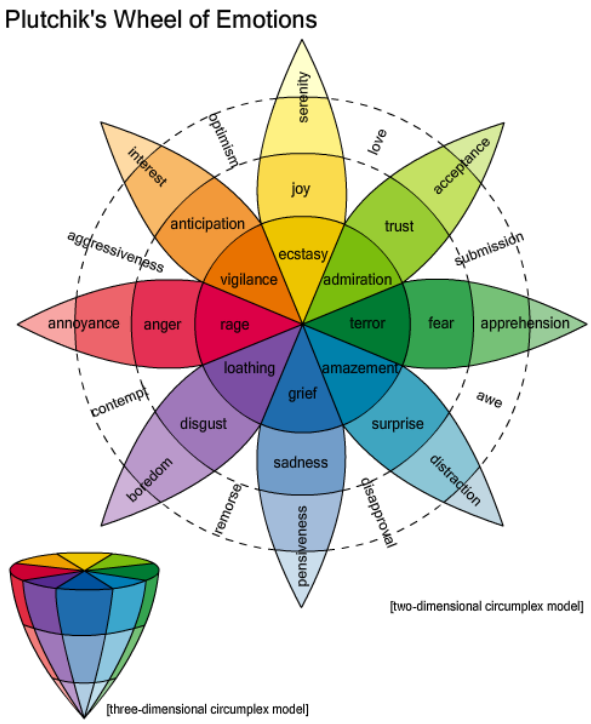

Figure 2.2: Plutchik's wheel of emotions (Mohammad and Turney, 2013a).

2.3.3 Intersection of Words Between Lexicon Resources

Bravo-Marquez et al. (2013) studied the intersection of words between some lexical resources: SentiWordNet (SWN3), NRC Lexicon, OpinionFinder (OPFIND), and AFINN. The number of common words between each pair of resources is shown in Table 2.1. SWN3 is clearly larger than the other resources.

	SWN3	**NRC**	**AFINN**	**OPFIND**
SWN3	147,306	x	x	x
NRC	13,634	14,182	x	x
AFINN	1,783	1,207	2,476	x
OPFIND	6,199	3,596	1,245	6,884
Distinct Words	149,114			

Table 2.1: Intersection of words between different Lexical Resources (Bravo-Marquez et al., 2013).

Nevertheless, each resource includes many neutral words provided by WordNet that lack useful information for the purpose of sentiment analysis purposes. Table 2.2 shows the overlap of words after discarding the neutral words from SentiWordNet, the neutral and mixed words from OpinionFinder and the words without emotion tags from NRC.

	SWN3	**NRC**	**AFINN**	**OPFIND**
SWN3	33,313	x	x	x
NRC	2,932	3,071	x	x
AFINN	1,203	721	1,871	x
OPFIND	3,703	1,658	900	4,311
Distinct Words	34,649			

Table 2.2: Intersection of non-neutral words (Bravo-Marquez et al., 2013).

The interaction of all the non-neutral words, can be better represented in the form of a Venn diagram shown in Figure 2.3.

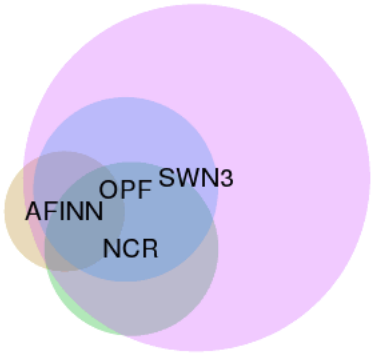

Figure 2.3: Intersections of words represented in a Venn diagram (Bravo-Marquez et al., 2013).

2.4 Cost-sensitive Analysis

Most supervised learning algorithms assume that all errors have the same cost. However this is not always the case. Some examples are:

- In credit, the cost of incorrectly giving credit is not the same as the loss of not giving credit to a good customer.
- In marketing, the cost of mailing a person that does not respond is usually smaller than not mailing a potential customer.
- In fraud detection the cost of useless investigation is not the same as loss of not investigating a real fraud.
- In medicine, the cost of not prescribing an exam to a sick person can be much higher than the cost of prescribing the exam to a healthy person.

The recognition that cost-sensitive analysis is important led to new research.

Breiman et al. (1984) and Elkan (2001) considered a different cost for different types of error. Núñez (1991) and Melville et al. (2005) analysed costs of tests, whose aim is to obtain attribute values, but did not consider misclassification costs. Freitas et al. (2007) considered both types of costs.

Although misclassification and test costs have been considered the most important types of costs, other types of costs exist. Turney (2000) created a taxonomy of the different types of cost that are involved in inductive concept learning. The author states that "*cost*" should be interpreted in an abstract sense, and can be measured in many different units, such as money (dollars, euros), temporal units (minutes, seconds), or other measures (e.g. measures of utility).

Some work has been done that considers more than one type of error. The first to do so was Turney (1995), but other authors followed (Zubek et al., 2004; Greiner et al., 2002; Chai et al., 2004; Ling et al., 2004).

Turney (1995) introduced a new algorithm for cost-sensitive classification, the ICET, that uses a genetic algorithm to evolve a population of biases for a decision

tree induction algorithm. Both cost of tests and cost of classification errors were considered.

An example of a real world application of cost-sensitive analysis is the work of Freitas et al. (2007) that applied cost-sensitive decision trees to medical data. The authors defined an algorithm for decision tree induction that considers costs, including test costs, delayed costs and costs associated with risk (economic and non-economical costs). Then they applied their strategy to train and evaluate cost-sensitive decision trees in medical data.

Chapter 3

Methodology: Determining Sentiment Value with Lexicons

Chapter 3 describes the data and the methods that have been used in this thesis. We start by presenting the text pre-processing techniques, as well as the lexicons that have been used in this work. Further on, we describe the sentiment classifier and different evaluation techniques that have been used.

3.1 Corpus Pre-processing

Unstructured textual data can be very noisy. Thus data cleaning is a very important step to achieve good results. The goal behind pre-processing is to prepare the data for the subsequent steps.

The following pre-processing techniques have been considered:

- **Removal of news without relevant information**
  Having empty documents or documents with irrelevant information only adds noise to the classification problem. Therefore, their removal is a very important task.

- **Conversion to lower case**

  This step consists on removing inconsistency on the use of upper and lower cases. Therefore, all the words were converted into lower case.

  This also makes the words form compatible with the lexicons used in the classification task. Moreover, as this task does not affect the meaning of the words, if it were not performed some words would not be considered to be the same word (e.g. *good* and *Good*) and that could affect negatively the results.

- **Stopwords removal**

  Stopwords are language-specific functional words. These are frequent words that do not add or remove any relevant information (i.e. prepositions pronouns, conjunctions). Some lists include about 400-500 stopwords for the English language. Examples include *a*, *but*, *if*, *or*.

  This process also allows the reduction of the corpus, leaving only essential words for the subsequent steps.

- **Spaces, punctuation and numbers removal**

  It is also important to remove unnecessary whitespaces, punctuation symbols and numbers.

- **Stemming**

  Solka et al. (2008) defines *stemming* as the process of removing suffixes and prefixes, leaving the root or stem of the word. The hypothesis is that words with a common stem or word root mostly describe similar meanings in text. For example:

  connect
  connected
  connecting

connection
connections

have a common stem *connect*.

As Porter (1980) stated, the performance of an information retrieval system will often improve if term groups such as these are conflated into a single term. This can be done by removing the various suffixes –ed, –ing, –ion, –ions to leave the single stem *connect*. Moreover, this process will reduce the total number of terms which is beneficial for many text mining operations.

The most commonly used stemmer is the *Porter Stemmer* (Porter, 1980).

3.2 Lexicon-based Approach

3.2.1 Sentiment Classifier

In this work, sentiment analysis was performed at document level (Section 2.2.1), that is, each news article was classified into positive, negative or neutral. A neutral classification means the article is nor good or bad for the company referred to in the article. We assume that each article refers to a single company.

The algorithm that was used to classify each news article proceeds as follows (see Algorithm 1):

- Each word of each document is classified into positive (if it is in the positive lexicon), negative (if it is in the negative lexicon) or neutral (if it is not in either of the two sentiment lexicons).
- If the sum of the number of positive words of a news article is larger than the sum of the number of negative words, than the document is classified as *positive*.

- If the sum of the number of negative words of a news article is larger than the sum of the number of positive words, than the document is classified as *negative*.
- If neither of the two previous conditions are satisfied, the news article is classified as *neutral*.

Algorithm 1 Document Sentiment Classification

```
procedure ClassifyDocumentSentiment(document)
    sum.positives = 0
    sum.negatives = 0
    for each word w in document do
        if sentiment(w) is positive then
            sum.positives = sum.positives + 1
        else if sentiment(w) is negative then
            sum.negatives = sum.negatives + 1
        end if
    end for
    if sum.positives > sum.negatives then
        return positive
    else if sum.positives < sum.negatives then
        return negative
    else
        return neutral
    end if
end procedure
```

3.2.2 Sentiment Lexicons

The information from the publicly available lexicon resources referred in Section 2.3.2 was downloaded and an extraction work was performed to obtain a list of positive and negative words. This work was carried out for each lexicon listed below.

- **OpinionFinder** (Wilson et al., 2005a)

  OpinionFinder (OF) classifies words in positive, negative and neutral. For the purpose of this thesis, only positive and negative words were considered.

  This results in a list of 2,718 positive words and 4,913 negative ones.

- **SentiWordNet** (Esuli and Sebastiani, 2006)

  As referred in Section 2.3.2, SentiWordNet (SWN) gives three numerical scores Obj(s), Pos(s), Neg(s).

  To classify a word into positive or negative an algorithm was developed (see Algorithm 2). It sums the positive classification of all synsets of a word and also sums the negative ones and divides them by the number of occurrences of the synset. If the difference between these two values is greater than threshold zero, then the word is classified as positive. If it is smaller, the word is classified as positive. If there is no difference between the two values, the word is ignored.

Algorithm 2 SentiWordNet Preparation

```
procedure ClassifyWordSentiment(word, threshold)
    if AvgPosSentiment(word) - AvgNegSentiment(word) > threshold then
        AddToPositiveLexicon(word)
    else if AvgNegSentiment(word) - AvgPosSentiment(word) > threshold then
        AddToNegativeLexicon(word)
    else
        ignore
    end if
end procedure
```

For example, if this algorithm is applied to the word *estimable* (Figure 2.1), it would be classified as positive, because:

- In synset 1, *estimable* is 0.75 positive and 0 negative.

- In synset 2, it is 0.625 positive and 0.25 negative.
- In synset 3, it is 0 positive and 0 negative.
- The average of the positive values is (0.75 + 0.625 + 0) / 3 = 0.458.
- The average of the negative values is calculated similarly and gives 0.083.
- As the average of positive values (0.458) is greater than the average of negative values (0.083), the word *estimable* is classified *positive*.

After applying this algorithm to SentiWordNet, the lexicon had 20,308 positive words and 17,597 negative ones.

- **AFINN** (Nielsen, 2011)

  As previously referred, AFFIN scoring ranges from -5 (very negative) to +5 (very positive). Therefore, to adapt this classification to a positive, negative or neutral classification, a word is classified as positive if it ranges from 1 to 5, as negative if it ranges from -1 to -5. Words with score 0 were ignored, as they will not help in the classification task.

  After this adaptation, the dictionary had 878 positive words and 1,598 negative ones.

- **NRC Lexicon** (Mohammad and Turney, 2013b)

  NRC Lexicon (NRC) classifies words in eight emotions (joy, trust, sadness, anger, surprise, fear, anticipation, disgust) and also tags them according to polarity classes: positive and negative.

  For this work only the positive and negative tags were considered, which resulted in 2,312 positive and 3,324 negative words.

3.3 Classifiers Evaluation

3.3.1 Evaluation Measures

A confusion matrix is a specific table layout that allows visualization of the performance of an algorithm (Vijayarani and Vinupriya, 2013). Each column of the matrix represents the predicted classes, while each row represents the actual classes (Table 3.1).

	Predicted Class A	Predicted Class B
Actual Class A	TP_A	FN_A
Actual Class B	FP_A	TN_A

Table 3.1: Confusion matrix (class A is the positive class).

The meaning of the four cells in the confusion matrix is:

- $\mathbf{TP}_A$ (True Positives) is the number of documents correctly assigned to class A.
- $\mathbf{TN}_A$ (True Negative) is the number of documents correctly assigned to class B.
- $\mathbf{FP}_A$ (False Positives) is the number of documents that are incorrectly assigned to class A. They belong to class B.
- $\mathbf{FN}_A$ (False Negatives) is the number of documents that belong to class A but were assigned to class B.

If class B is considered to be the positive class, the confusion matrix is as follows:

	Predicted Class B	Predicted Class A
Actual Class B	TP_B	FN_B
Actual Class A	FP_B	TN_B

Table 3.2: Confusion matrix (class B is the positive class).

Precision and recall can be derived from the confusion matrix:

Precision – Number of correctly labeled cases divided by the number of all returned cases:

$$p_i = \frac{TP_i}{TP_i + FP_i} \tag{3.1}$$

The formula above can be used to calculate the precision of both classes (i = A or B).

Recall – Number of correctly labeled cases divided by the number of cases that should have been returned:

$$r = \frac{TP_i}{TP_i + FN_i} \tag{3.2}$$

To evaluate the performance of each class, the F-measure metric *F1* can be used, which is defined as the harmonic mean of precision *(p_i)* and recall *(r_i)*. It gives equal weight to each document classification (Forman, 2003).

$$F1_i = 2 \times \frac{p_i \times r_i}{p_i + r_i} \tag{3.3}$$

In situations where are two classes, we have two values of *F1* (e.g. $F1_A$ and $F1_B$). These can be combined to obtain either *Micro F1* or *Macro F1*.

Micro-averaged F1 measure

The measures of precision and recall were adapted to consider more than two classes:

$$P = \frac{\sum_{i=1}^{M} TP_i}{\sum_{i=1}^{M} (TP_i + FP_i)} \tag{3.4}$$

$$R = \frac{\sum_{i=1}^{M} TP_i}{\sum_{i=1}^{M} (TP_i + FN_i)} \tag{3.5}$$

Then *Micro F1* is calculated as follows:

$$Micro\ \ F1 = 2 \times \frac{P \times R}{P + R} \tag{3.6}$$

where M is the number of classes.

Macro-averaged F1 measure

Macro-averaged F1 measure is defined as arithmetic mean of the F-measure for each class. It gives equal weight to each class (Forman, 2003).

$$Macro\, F1 = \frac{\sum_{i=1}^{M} F1_i}{M}, \qquad F1_i = 2 \times \frac{p_i \times r_i}{p_i + r_i} \tag{3.7}$$

where M is the number of classes.

3.3.2 Cost-sensitive Analysis

The evaluation measures just described fail to distinguish how grave the error is. As here we are dealing with ordinal data, we note that classifying a positive news as *neutral* is not as bad as classifying it as *negative*.

	Predicted Negative	Predicted Neutral	Predicted Positive
Actual Negative	0	0.5	1
Actual Neutral	0.5	0	0.5
Actual Positive	1	0.5	0

Figure 3.1: Cost matrix $\mathbb{C}\mathbf{ost}$ considered in this case study.

Analysing the cost matrix on Figure 3.1 it can be verified that:

- If a news article is correctly classified there is no costs.
- If a positive or negative news is classified as neutral a cost of 0.5 is applied. The same cost is used if a neutral news article is classified as positive or negative.
- If a positive news article is classified into the negative class or vice-versa, then the cost of 1 is applied, making this the most costly error.

The cost-sensitive analysis requires also a confusion matrix. The confusion matrix that results from the use of Algorithm 1 is shown in Table 3.3.

	Predicted Negative	Predicted Neutral	Predicted Positive
Actual Negative	200	75	40
Actual Neutral	30	100	15
Actual Positive	35	20	200

Table 3.3: Example of a confusion matrix $\mathbb{C}\mathbf{onf}$ (absolute frequency).

To normalise the data, all the values of the confusion are divided by the total number of cases. The new confusion matrix consists now of relative frequencies (see Figure 3.2).

	Predicted Negative	Predicted Neutral	Predicted Positive
Actual Negative	0 0.280	0.105	0.056 1
Actual Neutral	-0.5 0.042	0.140	0.021 0.5
Actual Positive	-1 0.049	0.028	0.280 0

Figure 3.2: Example of a confusion matrix $\mathbb{C}$**conf**$\mathbb{R}$ (relative frequency).

The success rate can be obtained easily from this matrix by summing up the relative frequencies in the diagonal. This results in 0.70 (0.28 + 0.14 + 0.28). The error rate is the complement of this, that is, 0.30. It is of course equal to the sum of all errors, that is, 0.105 + 0.056 + 0.042 + 0.021 + 0.049 + 0.028.

The confusion matrix $\mathbb{C}$**conf**$\mathbb{R}$ is then multiplied by the cost matrix $\mathbb{C}$**ost** (multiplication element by element). The result is shown in Table 3.4).

	Predicted Negative	Predicted Neutral	Predicted Positive
Actual Negative	0.000	0.052	0.056
Actual Neutral	0.021	0.000	0.010
Actual Positive	0.049	0.014	0.000

Table 3.4: Example of confusion matrix after applying costs.

This matrix is useful, as it provides different types of useful information. For instance, if some case is predicted positive, the probability that this is right is high, although there is some probability that an error can occur and the cost will be 0.056 + 0.010 = 0.066.

A plot that shows the cost distribution is then generated.

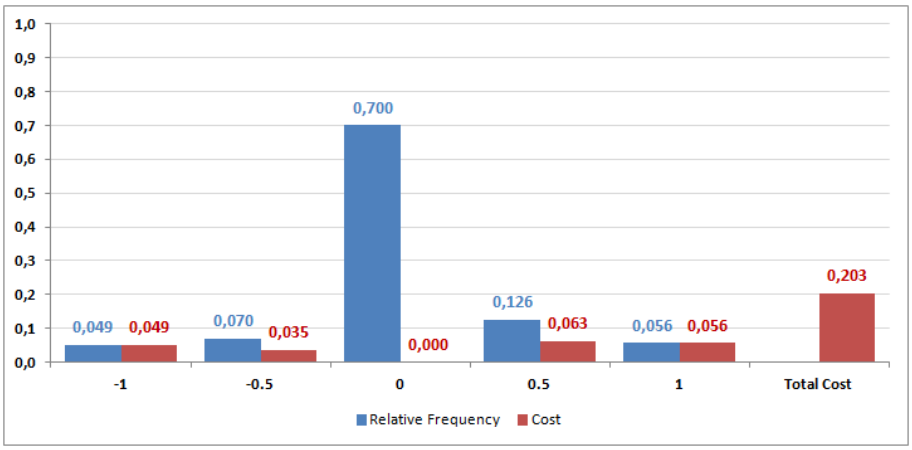

Figure 3.3: Example of a cost analysis plot.

Analysing Figure 3.3, we note:

- The number on x axis refers to different values of cost that appear in Figure 3.2. For instance, the value 0 appears in the diagonal.
- The blue bar refers to relative frequencies in $\mathbb{C}\mathbf{onf}\mathbb{R}$ matrix. This means that, for example, diagonal 0 (correctly classified news articles) has 70% of the cases.
- The red bar refers to the results after applying costs (Table 3.4). As expected the diagonal 0 has no costs associated.
- The last red bar on the right shows the total cost, that is, the sum of all costs. In our example above the total cost is 0.203.

In the next chapter, we apply these methods to financial news.

Chapter 4

Case Study Results

This chapter describes our case study. We present the data that was used, as well as the results of sentiment analysis for a series of studies that involve financial news.

4.1 Data

For this study 2,948 financial news articles were collected. They have been released between February 24th, 2014 and February 2nd, 2015.

Example of a news article:

WASHINGTON, Sept 16 (Reuters) - Boeing Co BA.N has won a large NASA contract to develop new "space taxis" that would fly astronauts to the International Space Station instead of relying on Russian spacecraft, an industry source said ahead of a NASA announcement expected on Tuesday.
The source said Boeing had received a full award for the multibillion-dollar contract, but financial details were not immediately available. NASA declined comment.
It was not immediately clear whether NASA would award smaller orders to rival bidders, including Space Exploration Technologies Corp, or SpaceX, and privately held Sierra Nevada Corp.
The contract has taken on new urgency in recent months, given escalating tensions

with Russia over its annexation of the Crimea region of Ukraine.

This news article was classified as *positive*.

The news articles did not have a sentiment classification tag. Consequently, the classification of articles into *positive*/*negative*/*neutral* was performed manually. Around 30% of the news (892 documents) were classified this way. Figure 4.1 shows the distribution of the manually classified news by sentiment class.

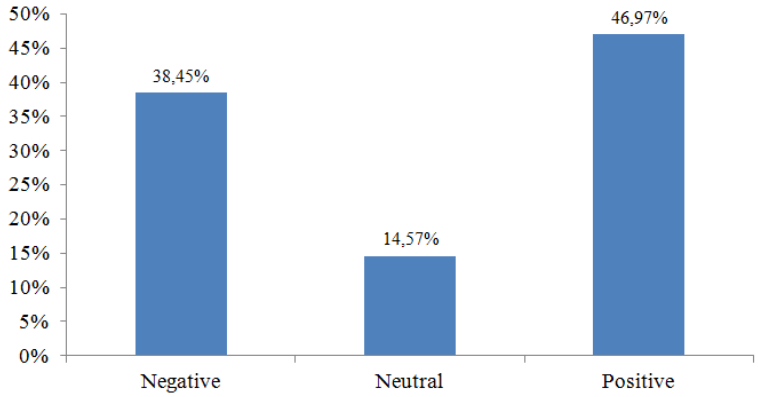

Figure 4.1: Sentiment distribution of the manually classified news.

Other authors have performed manual classification for different subjects, such as *starred movie (Turney, 2002; Pang et al., 2002), or* product reviews (Dave et al., 2003).

4.1.1 Programming Tools

Open source R software was used in this dissertation to compute all the steps involved in sentiment analysis. As Meyer et al. (2008) pointed out, *R has proven over the years to be one of the most versatile statistical computing environments available, and offers a battery of both standard and state of the art methodology.* Therefore, this appears to be a good choice.

R has several text mining packages available that facilitated the development of

this case study. Some of them are: *tm* (Feinerer, 2007) and *SnowballC* (Bouchet-Valat, 2013).

4.2 Corpus Pre-processing

As previously mentioned, many text mining tasks require that the text is pre-processed. The following pre-processing tasks were performed:

- **Removal of news without relevant information**

  Some retrieved news were empty or just included "NA" inside the file. After removing these news the data consisted of 2,885 news articles.

- **Conversion to lower case**

  All text was converted to lower case.

- **Stopwords removal**

  The list of stopwords considered is the one included in R's *tm* package. However, some words that were on the list were also in lexicons that were retrieved for this case study. Therefore these words were removed and they are listed in Appendix A. Some examples of words in this list are: *against*, *not* and *down*.

- **Spaces, punctuation and numbers removal**

  We follow the common approach and removed unnecessary whitespaces, punctuation symbols and numbers.

- **Stemming** This step was performed using the SnowballC package from R software. Porter Stemmer (Porter, 1980) was used for this task.

4.3 Overview of the Experiment and Results

In this section we present an overview of the experiments that were carried out.

In the first experiment (Section 4.4) publicly available lexicons (*Opinion Lexicon*, *OpinionFinder*, *SentiWordNet*, *AFINN* and *NRC*) were employed and used in the news sentiment classification. As *stemming* does not always improve the classification results, we carried out a study to verify if that was the case. Moreover, we adjusted the SentiWordNet list of words.

Next, we have merged some of these lexicons to see if the results could be improved (Section 4.5).

Additionally, a negation handling technique developed by Pang et al. (2002) was applied to the news (Section 4.6). The results were inconsistent, as the use of some lexicons improved, some got worst, and some had no change to its performance.

Finally, all the lexicons were enriched with more words from the financial world (Section 4.7). This study led to very positive results.

All experiments were evaluated using performance evaluation measures that were appropriate for classification (e.g. *Micro F1*). We also adopted a cost-sensitive analysis, where different costs were applied to different types of error.

4.4 Using Publicly Available Lexicons

In this section, the sentiment classification was carried out as described in Chapter 3. All documents (news) were processed with different sentiment lexicons discussed earlier.

4.4.1 Evaluation of Performance

Analysing the Effect of Stemming on Performance

Some authors (Bilotti et al., 2004; Harman, 1991; de Klerk, 2006) stated that stemming can decrease the performance of classifiers. Therefore an experiment was made to verify whether this was the case in this study.

The results are presented in Table 4.1 below.

				Negative class			Neutral class			Positive class		
	Stem	*Micro F1*	*Macro F1*	*p*	*r*	*F1*	*p*	*r*	*F1*	*p*	*r*	*F1*
AFINN	Yes	0.570	0.511	0.784	0.499	0.610	0.250	0.331	0.285	0.586	0.702	0.638
AFINN	No	**0.574**	0.517	0.795	0.510	0.622	0.253	0.338	0.289	0.588	0.699	0.639
NRC	Yes	**0.521**	0.401	0.713	0.268	0.390	0.212	0.138	0.167	0.524	0.847	0.647
NRC	No	0.518	0.390	0.699	0.271	0.391	0.175	0.108	0.133	0.523	0.847	0.647
OFinder	Yes	**0.509**	0.441	0.700	0.429	0.532	0.187	0.223	0.204	0.528	0.663	0.588
OFinder	No	0.507	0.439	0.690	0.423	0.524	0.190	0.223	0.205	0.526	0.663	0.586
OLex	Yes	**0.553**	0.514	0.671	0.676	0.673	0.253	0.423	0.317	0.626	0.492	0.551
OLex	No	0.548	0.510	0.666	0.673	0.670	0.256	0.423	0.319	0.615	0.484	0.542
SWN	Yes	**0.408**	**0.311**	0.392	0.292	0.334	0.174	0.062	0.091	0.433	0.611	0.507
SWN	No	**0.408**	0.308	0.393	0.294	0.337	0.149	0.054	0.079	0.435	0.611	0.508

Table 4.1: Classification results using a lexicon-based approach (with and without stemming).

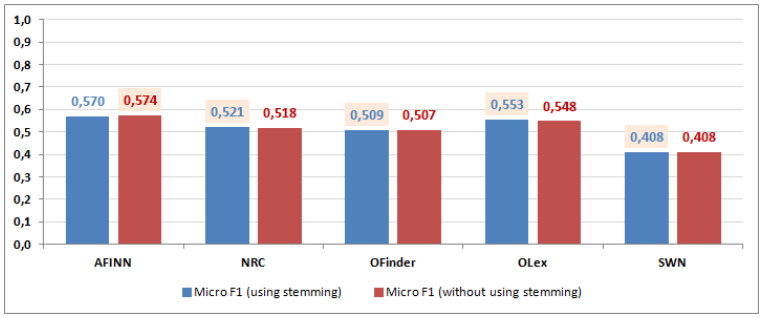

Figure 4.2: Evaluation results of the classifier using a lexicon-based approach (with and without stemming).

From the analysis of the previous results it can be concluded that stemming improves the overall performance of classifiers, with the exception of the AFINN

classifier. Therefore, we have decided to apply stemming in further experiments in this work.

Additionally, it can be verified in Table 4.1 that SentiWordNet had much worse performance than all the other lexicons. However, this result can be improved, as is shown next.

Improving SentiWordNet

Earlier (in Section 3.2.2), we have described an algorithm that transforms the SentiWordNet original classification into *positive*, *neutral* or *negative* values (Algorithm 2).

In the first experiment, the threshold used was zero, which has the effect that it classifies not strongly positive (negative) words as positive (negative). Consider, for instance, the word *academically*. SentiWordNet classifies it as 0.125 positive and 0 negative. But is it positive enough to be added to the positive word list? To answer this question we have carried out experiments with different thresholds in Algorithm 2. The results are shown in Table 4.2. The corresponding graph is shown in Figure 4.3.

			Negative class			Neutral class			Positive class		
Threshold	*Micro F1*	*Macro F1*	*p*	*r*	*F1*	*p*	*r*	*F1*	*p*	*r*	*F1*
0.0	0.408	0.311	0.392	0.292	0.334	0.174	0.062	0.091	0.433	0.611	0.507
0.1	0.460	0.385	0.493	0.717	0.584	0.173	0.146	0.158	0.512	0.346	0.413
0.2	0.476	0.423	0.505	0.810	0.623	0.274	0.346	0.306	0.573	0.243	0.342
0.3	**0.489**	**0.471**	**0.615**	0.569	0.591	0.249	0.554	0.344	**0.591**	0.403	0.479
0.5	0.406	0.376	0.590	0.662	0.624	0.212	0.685	0.324	0.523	0.110	0.181

Table 4.2: Classification results with SentiWordNet using different thresholds.

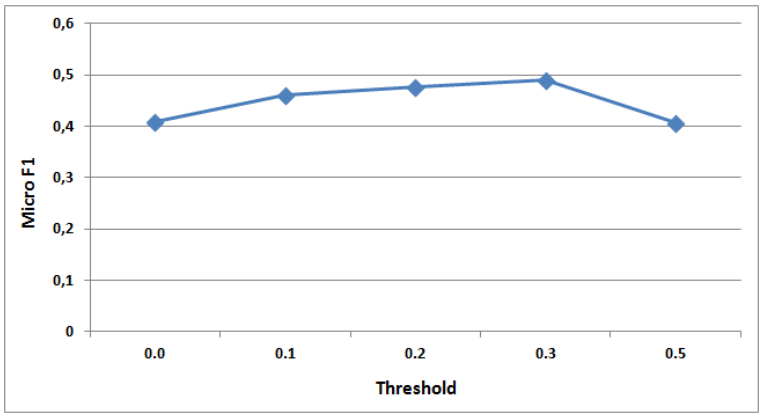

Figure 4.3: Graphical representation of classification results for *Micro F1* with SentiWordNet using different thresholds.

After analysing Table 4.2 and Figure 4.3, it can be concluded that thresholds 0.1, 0.2 and 0.3 greatly improve the results obtained with the zero threshold. The 0.3 threshold has the best *micro-averaged F1*, as precision of the positive and the negative class are higher for this threshold.

Therefore, in subsequent tests, the 0.3 threshold was used with SentiWordNet lexicon.

The transformed SentiWordNet lexicon includes 7,656 positive words and 4,690 negative ones.

Comparisons of Results

Table 4.3 shows the classification results with 5 different lexicons. In all cases we have used stemming and transformed SentiWordNet with 0.3 threshold.

			Negative class			Neutral class			Positive class		
	Micro F1	*Macro F1*	*p*	*r*	*F1*	*p*	*r*	*F1*	*p*	*r*	*F1*
AFINN	**0.570**	0.511	**0.784**	0.499	0.610	0.250	0.331	0.285	0.586	0.702	0.638
NRC	0.521	0.401	0.713	0.268	0.390	0.212	0.138	0.167	0.524	0.847	0.647
OFinder	0.509	0.441	0.700	0.429	0.532	0.187	0.223	0.204	0.528	0.663	0.588
OLex	**0.553**	0.514	0.671	0.676	0.673	0.253	0.423	0.317	**0.626**	0.492	0.551
SWN	0.489	0.471	0.615	0.569	0.591	0.249	0.554	0.344	0.591	0.403	0.479

Table 4.3: Classification results using a lexicon-based approach (with stemming).

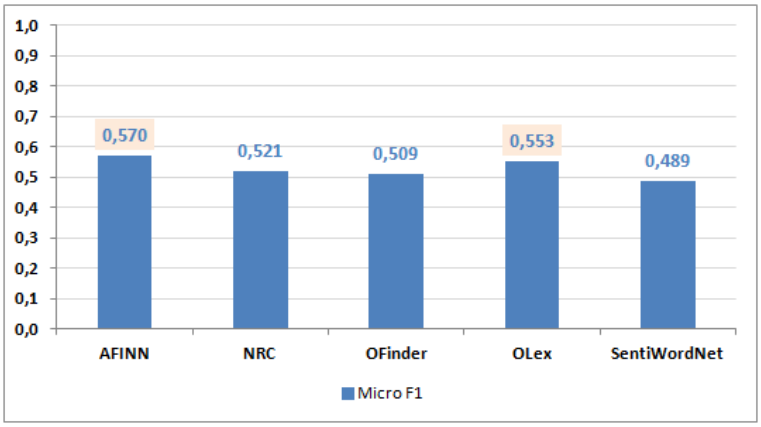

Figure 4.4: *Micro F1* of the publicly available lexicons.

Analysing Table 4.3 and Figure 4.4, it can be concluded that AFINN has the best overall performance, with the highest *micro-averaged F1* (57.0%). Opinion Lexicon is a close second. We note that AFINN has the best precision for the negative class, but Opinion Lexicon has the best precision for the positive class.

4.4.2 Cost-sensitive Analysis

As referred to earlier (in Section 3.3.2), when analysing ordinal data, attributing a uniform cost to all errors may not represent the best solution. These methods consider that classifying a positive news article as a negative or as a neutral class does not make a difference. However, classifying it as negative is worse than classifying it as neutral.

For these reasons, we have adopted a cost-sensitive analysis and used the follow-

ing cost matrix (showed already earlier).

	Predicted Negative	Predicted Neutral	Predicted Positive
Actual Negative	0	0.5	1
Actual Neutral	0.5	0	0.5
Actual Positive	1	0.5	0

Table 4.4: Cost matrix $\mathbb{C}$**ost** considered in this case study.

Using the methodology described earlier (Section 3.3.2), the following results were obtained (see Figure 4.5).

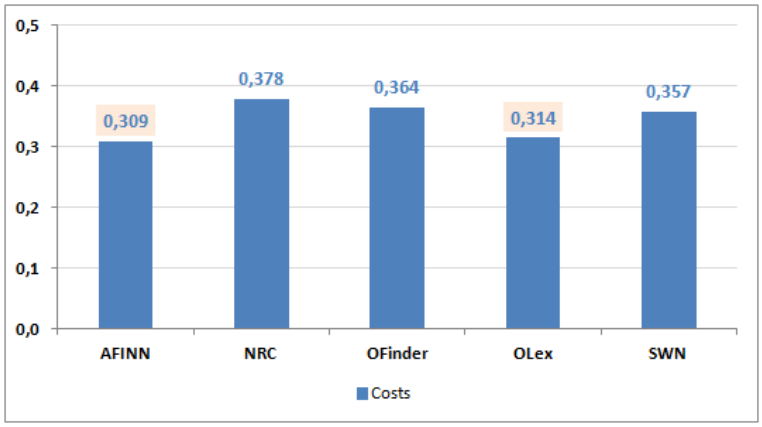

Figure 4.5: Cost analysis of the publicly available lexicons.

More details concerning cost analysis results for each lexicon are in Appendix B, Section B.1.

The main conclusions obtained from analysing the previous results are:

- AFINN and Opinion Lexicon have the best results having the lowest total cost of 0.309 and 0.314 respectively. These two lexicons have already been the identified as the ones with best performance when using *Micro F1* evaluation measure.

- SentiWordNet had the worst performance when using *Micro F1*. However, when analysing costs it jumped ahead of NRC and OpinionFinder. These had a large amount of negative news classified as positive (Figure B.1 in Appendix B).

As sentiment classification can be seen as a problem of classifying ordinal data, we consider that an evaluation using cost-sensitive analysis is appropriate. Therefore, in the following sections we present the cost-sensitive analysis results. The evaluation using usual performance evaluation measures (e.g *Micro F1*) can be consulted in Appendix C.

4.5 Merging Lexicons

Similarly as (Bravo-Marquez et al., 2013), we were interested to see whether merging different lexicons would improve the performance of the sentiment classification

The merged lexicons were the following:

1. **AFINN, NRC, OpinionFinder, Opinion Lexicon and SentiWordNet**

   In the first study we merged all the lexicons used individually in this work. This resulted on a lexicon with 8,374 positive words and 13,526 negative words.

2. **AFINN and Opinion Lexicon**

   In the next experiment we merged the two lexicons that had the best results: AFINN and Opinion Lexicon. The resulting lexicon had 2,451 positive words and 5,516 negative words.

The final experiment continued with the combination discussed above, that is, AFINN and Opinion Lexicon and added an extra lexicon from the set NRC, OpinionFinder and SentiWordNet.

3. **AFINN, Opinion Lexicon and NRC**

   This lexicon has 3,982 positive words and 7,032 negative words.

4. **AFINN, Opinion Lexicon and OpinionFinder**

   This lexicon has 3,114 positive words and 5,814 negative words.

5. **AFINN, Opinion Lexicon and SentiWordNet**

   This lexicon has 6,564 positive words and 12,022 negative words.

4.5.1 Cost-sensitive Analysis

In this section, we present the results after merging different lexicons. The results are shown in Figure 4.6.

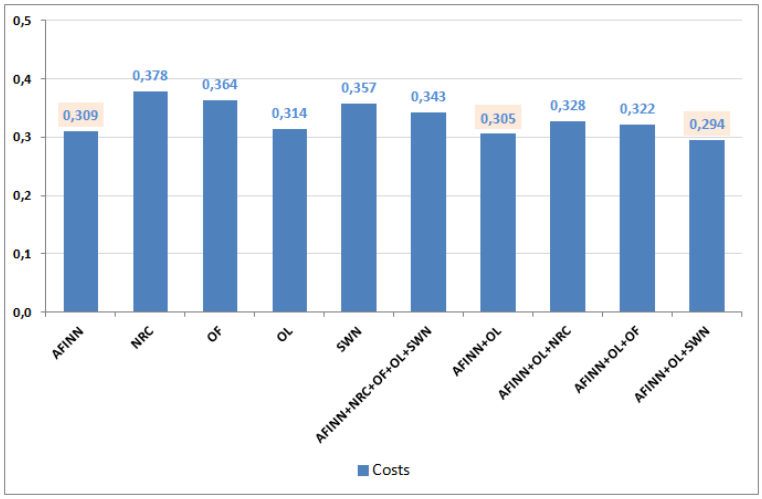

Figure 4.6: Cost analysis of the merged lexicons.

Analysing Figure 4.6, it can be concluded that merging AFINN, Opinion Lexicon and SentiWordNet obtains the best result with a total cost of 0.294. This lexicon had already performed this well while using *F1* measure. This combination has better results than using AFINN alone (the lexicon that had the best result while

analysing publicly available lexicons).

A more detailed cost analysis results of each merged lexicon is available in Appendix B, Section B.2.

An example of a news article that was incorrectly classified using AFINN (the best performing lexicon of the publicly available lexicons) and is now correctly classified using the merge of AFINN, Opinion Lexicon and SentiWordNet (the best performing lexicon of the merged lexicons) is the following:

Feb 25 (Reuters) - GE GE.N :

⋆ *Launches new distributed power business, announces $1.4 billion investment to meet world's need for on-site power.*

⋆ *Says GE targets global energy shift to* ***faster****, more* ***affordable*** *and* ***efficient*** *on-site power.*

⋆ *Says GE white paper predicts distributed power will grow 40 percent* ***faster*** *than global electricity* ***demand*** *between now and 2020.*

Analysing all the words on the news article, it can be verified that AFINN lexicon only contained the word "demand", that was on its negative words list. Therefore the news was incorrectly classified as negative.

However, if the merge of AFINN, Opinion Lexicon and SentiWordNet is used, three positive words are considered: "faster", "affordable" and "efficient". Moreover, the implemented classification algorithm uses a term frequency approach (Section 2.1.2), in which values reflect the number of occurrences of a term. As the word "faster" occurs two times, it increases the number of positive occurrences to four. The only negative word is still "demand". Therefore, the news article is correctly classified as positive.

4.6 Negation handling

The following study incorporates the technique developed by Pang et al. (2002) discussed in Section 2.1.4. It consists on adding the tag NOT_ to every word between the negation word (e.g. *not*, *isn't*, *didn't*) and the first punctuation mark following the negation word. Following this strategy, the following changes were made to all lexicons:

- Every word of the lexicons originated in a new word with the prefix *NOT_* (e.g. *love* originates *NOT_love*).
- If the word belonged to the positive list of words (e.g. *love*), the word with the tag NOT_ (e.g *NOT_love*) was added to the negative list.
- If the word belonged to the negative list of words (e.g. *hate*), the word with the tag NOT_ (e.g *NOT_hate*) was added to the positive list.

The advantage of this approach is that the same word originates in two separate terms, one for the plain occurrence, and other for the occurrence with negation.

However, Pang et al. (2002) reported that this strategy had a negligible, and on average slightly harmful effect on performance. Nevertheless, we have applied this technique to handle negation and evaluated its impact on the sentiment classification.

4.6.1 Cost-sensitive Analysis

In this section we present the cost-sensitive analysis with the aim to verify the impact of applying the negation technique.

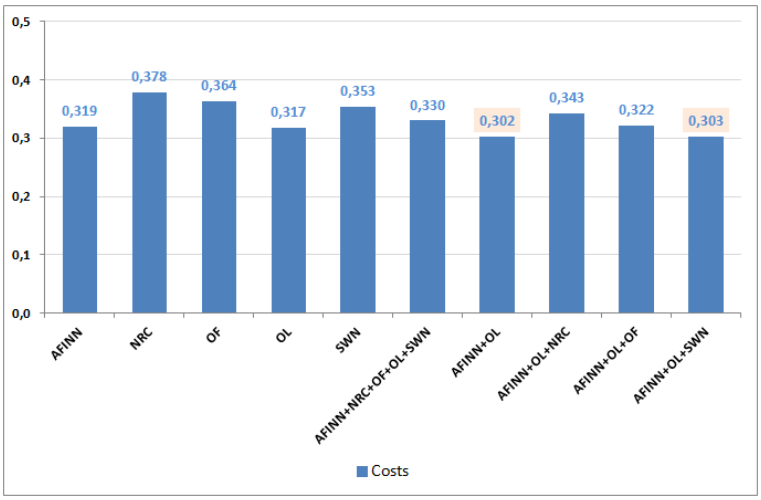

Figure 4.7: Cost analysis after applying negation technique.

	Total cost without applying negation	Total cost after applying negation	Improvement
AFINN	**0.309**	0.319	-0.010
NRC	0.378	0.378	0.000
OF	0.364	0.364	0.000
OL	**0.314**	0.317	-0.003
SWN	0.357	**0.353**	0.004
AFINN+NRC+OF+OL+SWN	0.343	**0.330**	0.013
AFINN+OL	0.305	**0.302**	0.003
AFINN+OL+NRC	**0.328**	0.343	-0.015
AFINN+OL+OF	0.322	0.322	0.000
AFINN+OL+SWN	**0.294**	0.303	-0.009

Table 4.5: Comparing cost before and after applying negation technique.

When analysing Figure 4.7 and Table 4.5 we can verify that negation handling improved the costs in some cases, but it had the opposite effect in others. As for the best combination identified earlier – AFINN + OL + SWN – negation handling resulted in slightly worse result. Nevertheless, this combination maintained its first place in the ranking.

A more detailed cost analysis of all lexicons can be consulted in Appendix B,

Section B.3.

Overall, it cannot be stated that treating negation is better or worse for performance, since there were different outcomes to different combinations of lexicons.

Moreover, in the cases that negation improved the performance, the improvement was negligible. Therefore, this technique is not used in this work from this moment on.

Below we present a news article that was incorrectly classified as positive by the lexicon AFFIN+OL+SWN, but after applying this negation treatment was correctly classified as negative (words that had their polarity inverted have the prefix NOT_ in grey):

(Adds further comments, background, ***share*** *price) PARIS, May 6 (Reuters) - French President Francois Hollande said General Electric's GE.N bid for Alstom's ALSO.PA energy business is* ***not*** not_ ***acceptable*** not_ *as* not_ *it* not_ *stands* not_ *and* not_ *that* not_ *the* not_ *government's* not_ *aim* not_ *is* not_ *to* not_ *get* ***not_*** ***better*** not_ *offers. "The bid is* ***not*** ***not_*** ***good*** ***not_*** ***enough****, it's* ***not*** ***not_*** ***acceptable****," Hollande told RMC radio on Tuesday. Asked whether it was possible that the state, which currently holds around 1 percent in Alstom, could itself* ***increase*** *its stake in the* ***ailing*** *engineering group, he said: "For now I would* ***prefer*** *to get* ***better*** *offers." Alstom said last week it was reviewing a binding $16.9 billion bid from GE for its energy arm, although it has* ***not*** not_ *turned* not_ *down* not_ *a* ***not_*** ***rival*** not_ *offer* not_ *from* not_ *Germany's* not_ *Siemens* not_ *SIEGn.DE . French Economy Minister Arnaud Montebourg also came out against the GE offer on Monday but opened the door for a deal that would also combine the two companies'* ***rail*** *businesses. "In its current form, we* ***unfortunately*** *cannot* not_ *give* ***not_*** ***backing*** not_ *to* not_ *the* not_ *proposals* not_ *that* not_ *you* not_ *have* not_ *made* not_ *based* not_ *solely* not_ *on* not_ *the* not_ *purchase* not_ *of* not_ *Alstom's* not_ *energy* not_ *activities," Montebourg*

wrote in a letter to GE Chief Executive Jeff Immelt. ID:nL6N0NR3QZ ***Shares*** *in Alstom were 1.1 percent lower at 29.03 euros by 0725 GMT, among the* ***worst*** *performers on a 0.2 percent* ***firmer*** *French blue-chip CAC 40 index .FCHI.*

Before applying the negation technique, this news article had 12 positive words and 9 negative words, which resulted in a positive classification.

After applying the negation technique, one word that was previously tagged as negative is now considered positive (*"rival"*) while 6 words that were previously tagged as positive are now considered negative (*"acceptable"*, *"better"*, *"good"*, *"enough"*, *"acceptable"*, *"backing"*). This results in 7 positive words and 14 negative words, changing the classification of the news article to *negative*.

4.7 Lexicon Enrichment

The accuracy of sentiment classification can be highly sensitive to the domain to which it is applied. Therefore, around 40 news were analysed with the aim to extract financial terms and assign the appropriate sentiment.

These news were randomly chosen from the list of news that had not been manually classified. If the news analysed were from the list of manually classified news it could lead to overfitting, that is, the classifier could fit the training set very well, but fail to replicate the result in future situations.

The terms chosen were assigned a positive or negative classification and added to the previously analysed lexicons. In total 21 positive words and 38 negative words were identified. The list of words added to the lexicons can be consulted in Appendix D. Some examples of terms that were added are:

- **Takeover** - A situation in which a company gets control of another company by buying enough of its shares (assigned a negative sentiment).

- **Subprime** - The practice of lending money, especially to buy a house, to people who may not be able to pay it back (assigned a negative sentiment).
- **Belt-tightening** - A reduction in spending by consumers, businesses, governments, etc., usually because they have financial problems (assigned a negative sentiment).
- **Dividend** - (A part of) the profit of a company that is paid to the people who own shares in it (assigned a positive sentiment).

Additionally, some words were removed from the lexicons, because of their specific meaning in the financial world that did not match the assigned sentiment classification. Some examples are:

- **Share** - Removed because in finance a share is a part of the company. Therefore it is not positive nor negative.
- **Indebted** - Removed from the list of positive words because in finance it means that it owes money. In general it may mean *grateful because of help given.* Moreover, this word was added to the negative list.

4.7.1 Cost-sensitive Analysis

The evaluation results of the classifiers after enriching the sentiment lexicons are shown in Figure 4.8.

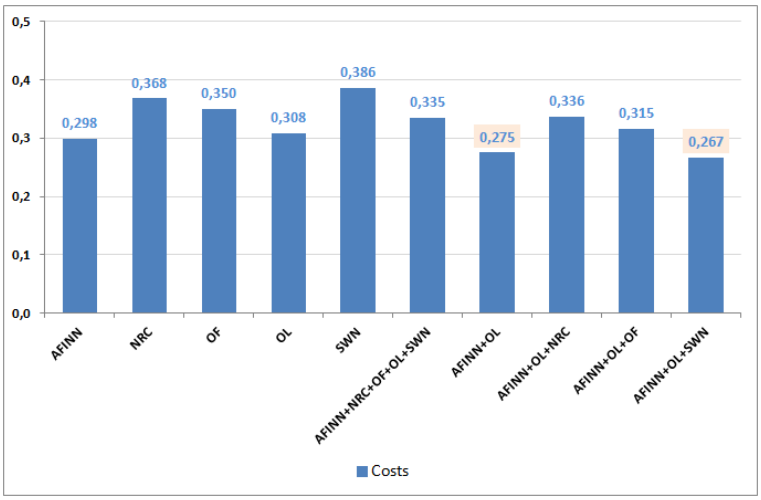

Figure 4.8: Cost analysis after lexicon enrichment.

	Total cost without lexicon enrichment	Total cost after lexicon enrichment	Improvement
AFINN	0.309	**0.298**	0.011
NRC	0.378	**0.368**	0.010
OF	0.364	**0.350**	0.014
OL	0.314	**0.308**	0.006
SWN	**0.357**	0.386	-0.029
AFINN+NRC+OF+OL+SWN	0.343	**0.335**	0.008
AFINN+OL	0.305	**0.275**	0.030
AFINN+OL+NRC	**0.328**	0.336	-0.008
AFINN+OL+OF	0.322	**0.315**	0.007
AFINN+OL+SWN	0.294	**0.267**	0.027

Table 4.6: Comparing cost before and after adding financial terms to lexicons.

Analysing Figure 4.8 and Table 4.6, it can be concluded that:

- In eight of the ten lexicons used, the cost analysis improved when new terms were added.
- The lexicon that previously obtained the best result, AFINN+OL+SWN, with a total cost of 0.294, got even better results with the incorporation of new terms in the lexicon, reducing its total cost to 0.267.

A more detailed cost analysis results of each lexicon is available in Appendix B, Section B.4.

An example of a news article that was previously incorrectly classified by the merge of AFINN, Opinion Lexicon and SentiWordNet and with the lexicon's enrichment is now correctly classified is the following:

April 17 (Reuters) - General Electric Co GE.N reported a ***decline*** *in quarterly net income on Thursday,* ***hurt*** *by lower revenue in its transportation business that sells locomotives, but the U.S. conglomerate's overall industrial profits rose by 12 percent. First-quarter net earnings fell to $3 billion, or 30 cents per* ***share****, from $3.53 billion, or 34 cents per* ***share****, a year ago, when the company's results were* ***boosted*** *by its sale of NBCUniversal. (Reporting by Lewis Krauskopf, Editing by Franklin Paul) ((lewis.krauskopf@thomsonreuters.com)(646-223-6082)) Keywords: GENERAL ELECTRIC RESULTS/*

This article is tagged as negative, however it was incorrectly classified as positive.

The words that were considered positive were *"share"* (occurred twice) and *"boosted"*, and the negative words were *"decline"* and *"hurt"*. With the lexicon enrichment this news article is now classified as negative. This happened because of the removal of the word *"share"* from the lexicon.

The results of this study confirm that the performance of sentiment classification is indeed highly sensitive to the domain to which it is applied. Therefore developing a lexicon oriented to the subject studied can be very beneficial to the sentiment classification task.

Chapter 5

Conclusions

5.1 Main Conclusions

In this thesis we described a system for automatic detection of sentiment in financial news. The goal was to develop a system that could help investors by filtering the news and identifying the items that are important and leaving out others. The sentiment value positive or negative (but excluding the neutral) was used as the indicator of importance. This can help the user with the impossible task of going through all the financial news that are published every day around the world. To achieve this goal, several studies were carried out with the intent of exploiting sentiment classification in this process and improving it.

Our system includes several pre-processing steps. First, it includes the pre-processing of the corpus: removal of text without relevant information; conversion of the text to lower case; stopwords removal; spaces, punctuation and numbers removal and stemming. Second, all the lexicons used in this work (*Opinion Lexicon*, *OpinionFinder*, *SentiWordNet*, *AFINN* and *NRC*) were adapted to have only two lists of words: *positive* and *negative*. These steps enabled us to carry out the subse-

quent studies.

The first one used publicly available lexicons to classify each news article. The predictions obtained were compared with the correct values and the results were satisfactory. However, there was room for improvement. So, we decided to merge some of the lexicons and repeated the evaluation. The results were very positive as in almost all cases this resulted in improved classification results.

Additionally, a technique developed by Pang et al. (2002) to handle negation was applied. All the words between the negation word (e.g. *not*, *isn't*, *didn't*) and the next punctuation mark were added the prefix $NOT\_$. These words had their polarity inverted, that is, if a positive word was added the prefix $NOT\_$ then it became negative, and vice-versa. However, this experiment had inconsistent results. The results with some lexicons improved, with others got worse, or else there was no change.

The final study had the objective to verify whether the accuracy of sentiment classification was indeed sensitive to the financial domain. Therefore, some words were added to the sentiment lexicons, mostly words with special meaning for finance (e.g. *dividend*, *takeover*, *subprime*). Moreover, some words were removed from the publicly available lexicons, as they had a different meaning in the financial world. Therefore, they did not belong to the positive or negative list (e.g. *share*). This experiment of enriching the lexicons had a very positive result, and led to improved classification results.

All the experiments carried out in this work were evaluated using performance evaluation measures (e.g. Micro F1). Moreover, we used a cost-sensitive analysis. This type of analysis applies different costs to different types of error. As we are dealing with ordinal data, this type of analysis is appropriate. Misclassifying a positive news as *negative* is worse than classifying it as *neutral*. The results of both evaluations were compared to verify whether they followed the same trends, which

was indeed the case.

5.2 Future Work

As the negation handling technique developed by Pang et al. (2002) did not lead to improved results, other approaches to negation handling could be tested. For instance, Hu and Liu (2004) and Grefenstette et al. (2004) implemented a limited scoping of negation to its following 5 words. This means that only the five words following the negation word are rewritten with a $NOT\_$ prefix.

Another possibility to improve this work is to use *lemmatization*. *Lemmatization* is similar to word *stemming*, but it does not generate a stem of the word. It replaces the suffix of a word with a typical word suffix to get the normalised word form. For example, the words *computes*, *computing*, *computed* would be stemmed to *comput*, but their normalized form is the infinitive of the verb: *compute* (Plisson et al., 2004).

In this study, we used a lexicon-based approached. However, a Machine Learning approach could be, perhaps, also a good choice. We could thus use, for example, *random forest*, *decision trees* or *neural networks* as models that could learn to classify texts into the three classes on the basis of pre-classified data.

Other improvement is in the direction of enriching further the existing lexicons. This study had very good results, but adding more financial words to the lexicons and removing words that are harmful for the classification results could lead to even better results.

To help investors better understand stock markets evolution and how news articles affect them, an analysis of the correlation of news sentiment and the stock prices could be carried out in future.

References

Ahmad, K., Cheng, D., and Almas, Y. (2006). Multi-lingual sentiment analysis of financial news streams. In *Proc. of the 1st Intl. Conf. on Grid in Finance*.

Batrinca, B. and Treleaven, P. C. (2014). Social media analytics: a survey of techniques, tools and platforms. *AI & SOCIETY*, pages 1–28.

Bilotti, M. W., Katz, B., and Lin, J. (2004). What works better for question answering: Stemming or morphological query expansion. In *Proceedings of the Information Retrieval for Question Answering (IR4QA) Workshop at SIGIR*, volume 2004, pages 1–3.

Bouchet-Valat, M. (2013). Snowballc: Snowball stemmers based on the c libstemmer utf-8 library.

Bradley, M. M. and Lang, P. J. (1999). Affective norms for english words (anew): Instruction manual and affective ratings. Technical report, Citeseer.

Bravo-Marquez, F., Mendoza, M., and Poblete, B. (2013). Combining strengths, emotions and polarities for boosting twitter sentiment analysis. In *Proceedings of the Second International Workshop on Issues of Sentiment Discovery and Opinion Mining*, page 2. ACM.

Breiman, L., Friedman, J., Stone, C. J., and Olshen, R. A. (1984). *Classification and regression trees*. CRC press.

Chai, X., Deng, L., Yang, Q., and Ling, C. X. (2004). Test-cost sensitive naive bayes classification. In *Data Mining, 2004. ICDM'04. Fourth IEEE International Conference on*, pages 51–58. IEEE.

Das, S. and Chen, M. (2001). Yahoo! for amazon: Extracting market sentiment from stock message boards. In *Proceedings of the Asia Pacific finance association annual conference (APFA)*, volume 35, page 43. Bangkok, Thailand.

Dave, K., Lawrence, S., and Pennock, D. M. (2003). Mining the peanut gallery: Opinion extraction and semantic classification of product reviews. In *Proceedings of the 12th international conference on World Wide Web*, pages 519–528. ACM.

de Klerk, A. (2006). Keyword identification for service-desk call classification. *B.Sc. thesis, University of Maastricht.*

Elkan, C. (2001). The foundations of cost-sensitive learning. In *International joint conference on artificial intelligence*, volume 17, pages 973–978. Citeseer.

Esuli, A. and Sebastiani, F. (2006). Sentiwordnet: A publicly available lexical resource for opinion mining. In *Proceedings of LREC*, volume 6, pages 417–422.

Feinerer, I. (2007). tm: Text mining package. r package version 0.3. *URL http://CRAN. R-project. org/package= tm.*

Feldman, R. (2013). Techniques and applications for sentiment analysis. *Communications of the ACM*, 56(4):82–89.

Forman, G. (2003). An extensive empirical study of feature selection metrics for text classification. *The Journal of machine learning research*, 3:1289–1305.

Freitas, A., Costa-Pereira, A., and Brazdil, P. (2007). Cost-sensitive decision trees applied to medical data. In *Data Warehousing and Knowledge Discovery*, pages 303–312. Springer.

Goldberg, A. B. and Zhu, X. (2006). Seeing stars when there aren't many stars: graph-based semi-supervised learning for sentiment categorization. In *Proceedings of the First Workshop on Graph Based Methods for Natural Language Processing*, pages 45–52. Association for Computational Linguistics.

Grefenstette, G., Qu, Y., Shanahan, J. G., and Evans, D. A. (2004). Coupling niche browsers and affect analysis for an opinion mining application. In *RIAO*, pages 186–194. Citeseer.

Greiner, R., Grove, A. J., and Roth, D. (2002). Learning cost-sensitive active classifiers. *Artificial Intelligence*, 139(2):137–174.

Harman, D. (1991). How effective is suffixing? *JASIS*, 42(1):7–15.

Hatzivassiloglou, V. and McKeown, K. R. (1997). Predicting the semantic orientation of adjectives. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics and Eighth Conference of the European Chapter of the Association for Computational Linguistics*, pages 174–181. Association for Computational Linguistics.

Hatzivassiloglou, V. and Wiebe, J. M. (2000). Effects of adjective orientation and gradability on sentence subjectivity. In *Proceedings of the 18th conference on Computational linguistics-Volume 1*, pages 299–305. Association for Computational Linguistics.

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

Jindal, N. and Liu, B. (2006). Identifying comparative sentences in text documents. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 244–251. ACM.

Kim, S.-M. and Hovy, E. (2004). Determining the sentiment of opinions. In *Proceedings of the 20th international conference on Computational Linguistics*, page 1367. Association for Computational Linguistics.

Koppel, M. and Shtrimberg, I. (2006). Good news or bad news? let the market decide. In *Computing attitude and affect in text: Theory and applications*, pages 297–301. Springer.

Ling, C. X., Yang, Q., Wang, J., and Zhang, S. (2004). Decision trees with minimal costs. In *Proceedings of the twenty-first international conference on Machine learning*, page 69. ACM.

Liu, B. (2010). Sentiment analysis and subjectivity. *Handbook of natural language processing*, 2:627–666.

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

Liu, B. and Zhang, L. (2012). A survey of opinion mining and sentiment analysis. In *Mining Text Data*, pages 415–463. Springer.

Melville, P., Provost, F., Saar-Tsechansky, M., and Mooney, R. (2005). Economical active feature-value acquisition through expected utility estimation. In *Proceedings of the 1st international workshop on Utility-based data mining*, pages 10–16. ACM.

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

Miller, G. A., Beckwith, R., Fellbaum, C., Gross, D., and Miller, K. J. (1990). Introduction to wordnet: An on-line lexical database*. *International journal of lexicography*, 3(4):235–244.

Mohammad, S. M. and Turney, P. D. (2013a). Crowdsourcing a word–emotion association lexicon. *Computational Intelligence*, 29(3):436–465.

Mohammad, S. M. and Turney, P. D. (2013b). Nrc emotion lexicon. Technical report, NRC Technical Report.

Nielsen, F. Å. (2011). A new anew: Evaluation of a word list for sentiment analysis in microblogs. *arXiv preprint arXiv:1103.2903*.

Núñez, M. (1991). The use of background knowledge in decision tree induction. *Machine learning*, 6(3):231–250.

Pang, B. and Lee, L. (2005). Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 115–124. Association for Computational Linguistics.

Pang, B. and Lee, L. (2008). Opinion mining and sentiment analysis. *Foundations and trends in information retrieval*, 2(1-2):1–135.

Pang, B., Lee, L., and Vaithyanathan, S. (2002). Thumbs up?: sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*, pages 79–86. Association for Computational Linguistics.

Plisson, J., Lavrac, N., Mladenic, D., et al. (2004). A rule based approach to word lemmatization. *Proceedings of IS-2004*, pages 83–86.

Porter, M. F. (1980). An algorithm for suffix stripping. *Program: electronic library and information systems*, 14(3):130–137.

Ruiz-Martínez, J. M., Valencia-García, R., and García-Sánchez, F. (2012). Semantic-based sentiment analysis in financial news. In *International Workshop on Finance and Economics on the Semantic Web (FEOSW 2012)*, page 38.

Santorini, B. (1990). Part-of-speech tagging guidelines for the penn treebank project (3rd revision).

Schumaker, R. P., Zhang, Y., Huang, C.-N., and Chen, H. (2012). Evaluating sentiment in financial news articles. *Decision Support Systems*, 53(3):458–464.

Solka, J. L. et al. (2008). Text data mining: theory and methods. *Statistics Surveys*, 2:94–112.

Taboada, M., Brooke, J., Tofiloski, M., Voll, K., and Stede, M. (2011). Lexicon-based methods for sentiment analysis. *Computational linguistics*, 37(2):267–307.

Tsur, O., Davidov, D., and Rappoport, A. (2010). Icwsm-a great catchy name: Semi-supervised recognition of sarcastic sentences in online product reviews. In *ICWSM*.

Turney, P. (2000). Types of cost in inductive concept learning.

Turney, P. D. (1995). Cost-sensitive classification: Empirical evaluation of a hybrid genetic decision tree induction algorithm. *Journal of artificial intelligence research*, pages 369–409.

Turney, P. D. (2002). Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 417–424. Association for Computational Linguistics.

Vijayarani, S. and Vinupriya, M. (2013). Performance analysis of canny and sobel edge detection algorithms in image mining. *Int. J. Innovative Res. Comp. Commun. Eng*, 1(8).

Wiebe, J., Wilson, T., Bruce, R., Bell, M., and Martin, M. (2004). Learning subjective language. *Computational linguistics*, 30(3):277–308.

Wilson, T., Hoffmann, P., Somasundaran, S., Kessler, J., Wiebe, J., Choi, Y., Cardie, C., Riloff, E., and Patwardhan, S. (2005a). Opinionfinder: A system for subjectivity analysis. In *Proceedings of hlt/emnlp on interactive demonstrations*, pages 34–35. Association for Computational Linguistics.

Wilson, T., Wiebe, J., and Hoffmann, P. (2005b). Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of the conference on human language technology and empirical methods in natural language processing*, pages 347–354. Association for Computational Linguistics.

Wilson, T., Wiebe, J., and Hwa, R. (2004). Just how mad are you? finding strong and weak opinion clauses. In *aaai*, volume 4, pages 761–769.

Zubek, V. B., Dietterich, T. G., et al. (2004). Pruning improves heuristic search for cost-sensitive learning. Technical report, Corvallis, OR: Oregon State University, Dept. of Computer Science.

Appendix A

Pre-processing - Stopwords

In this section we present the words that were removed from the list of stopwords of the *tm* package.

Stopwords removed	
a	not
about	off
above	on
against	only
all	other
am	out
an	over
as	same
be	some
by	such
do	then
down	through
further	too
have	under
i	up
no	very

Table A.1: Removed stopwords

Appendix B

Cost Analysis

B.1 Publicly Available Lexicons

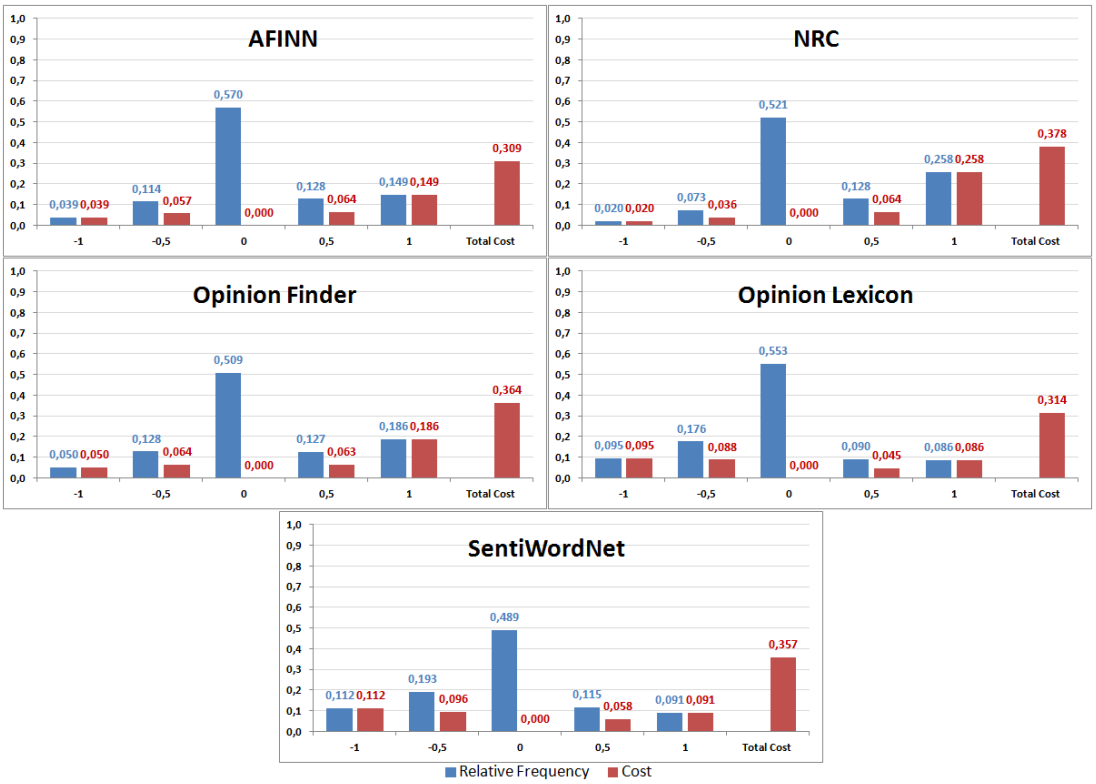

Figure B.1: Detailed cost analysis of the publicly available lexicons.

B.2 Merged Lexicons

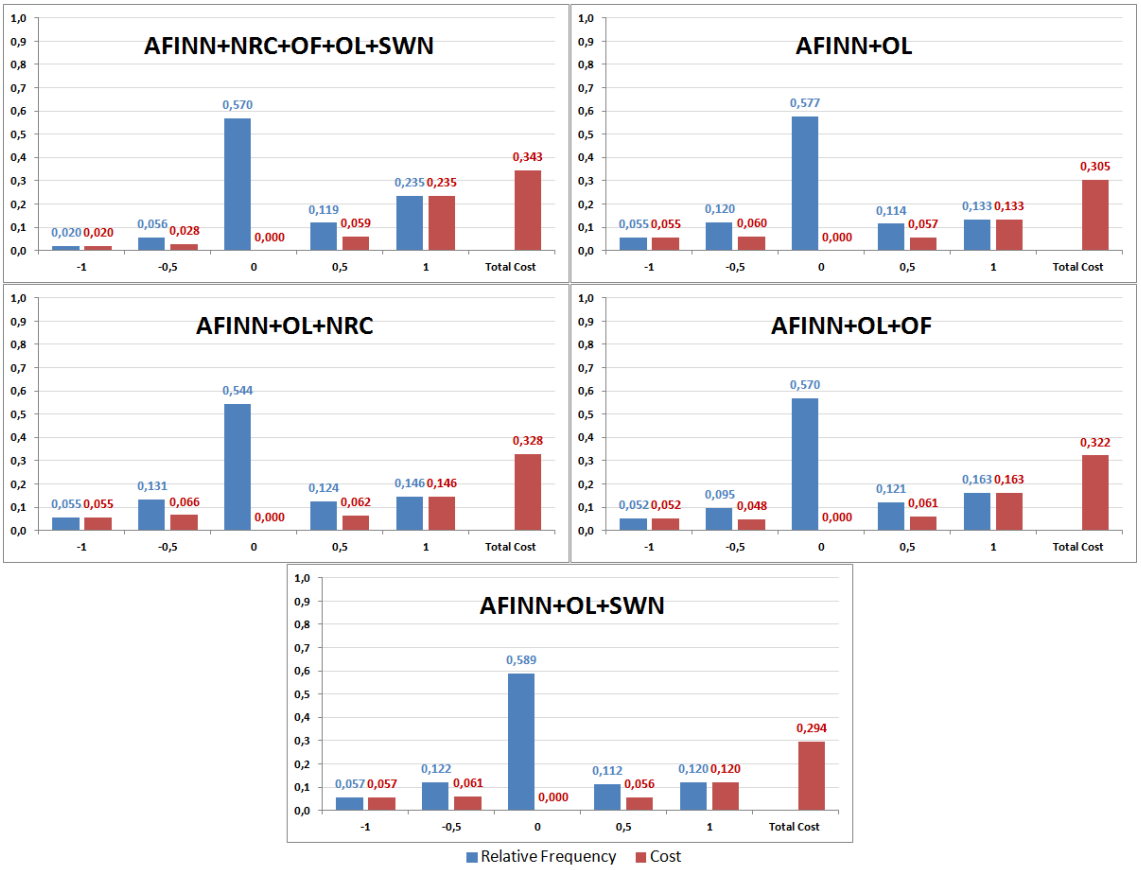

Figure B.2: Detailed cost analysis of the merged lexicons.

B.3 Negation Handling

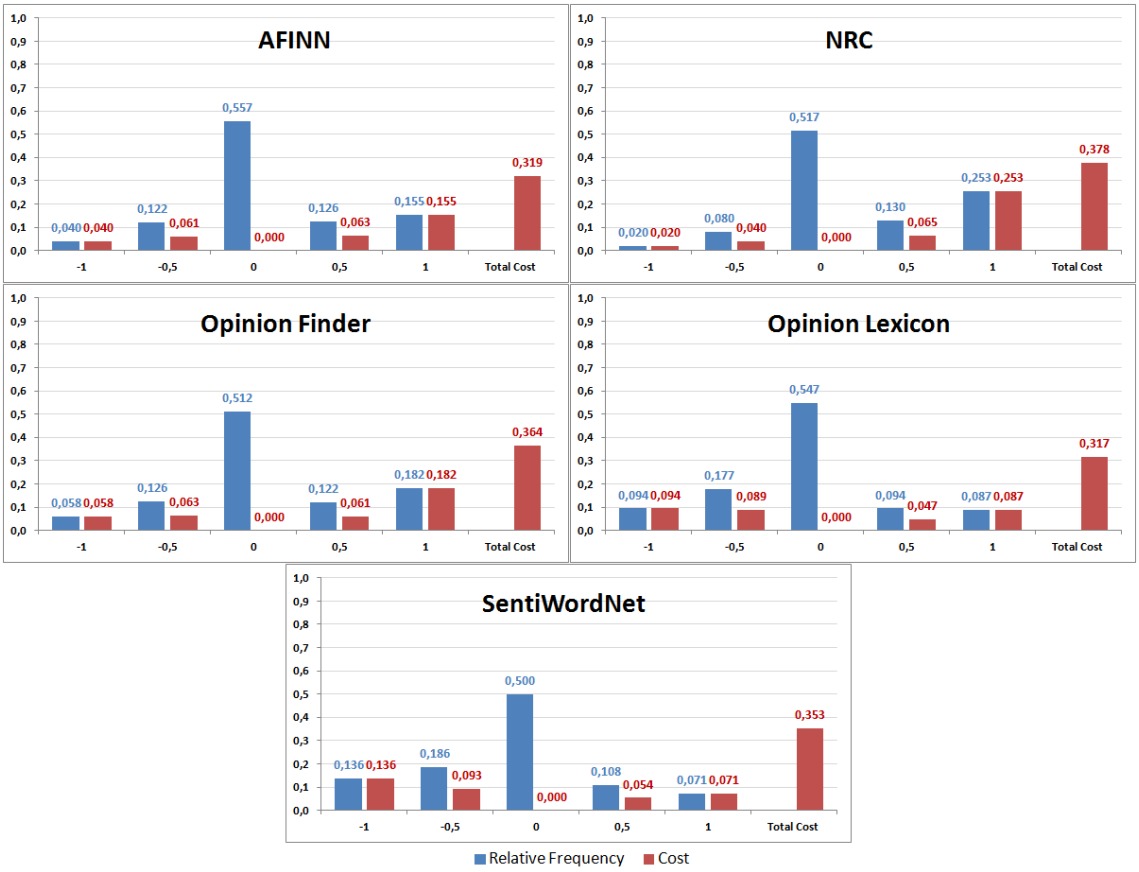

Figure B.3: Detailed cost analysis of the publicly available lexicons after applying negation technique.

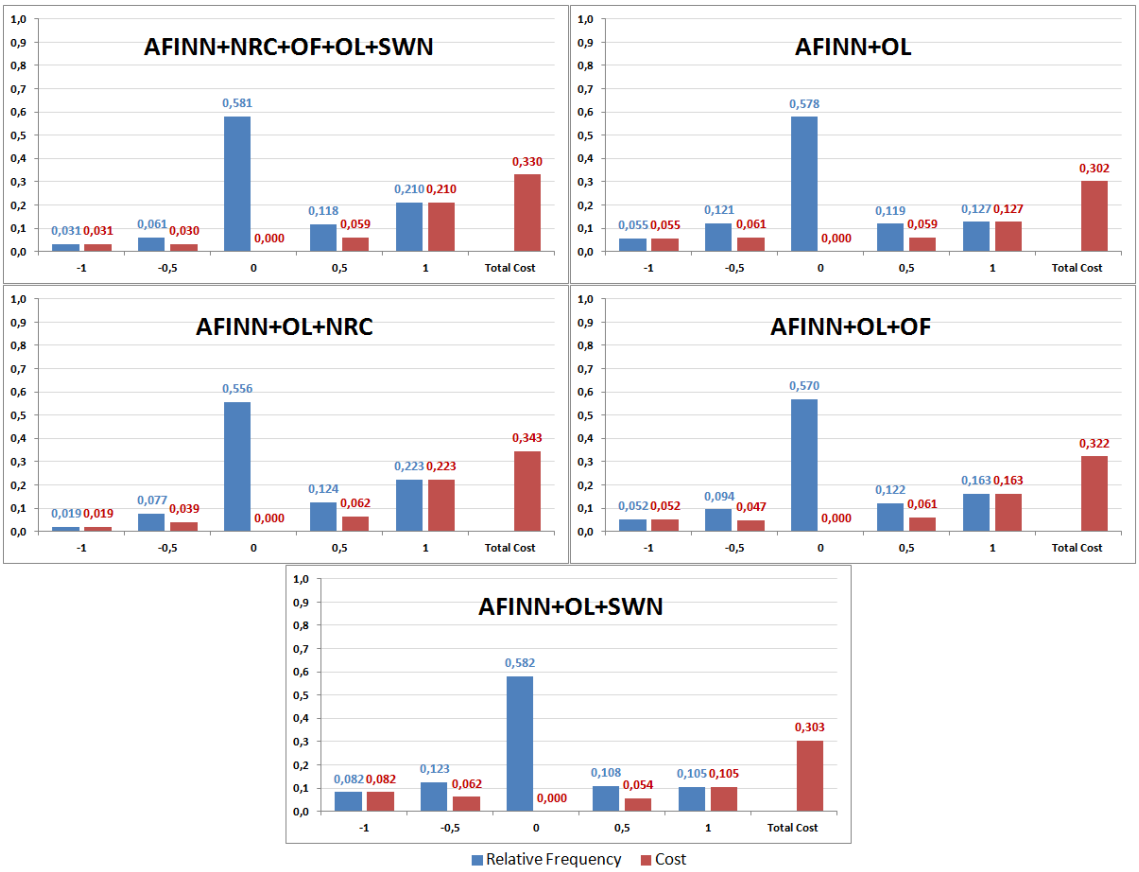

Figure B.4: Detailed cost analysis of the merged lexicons after applying negation technique.

B.4 Lexicon Enrichment

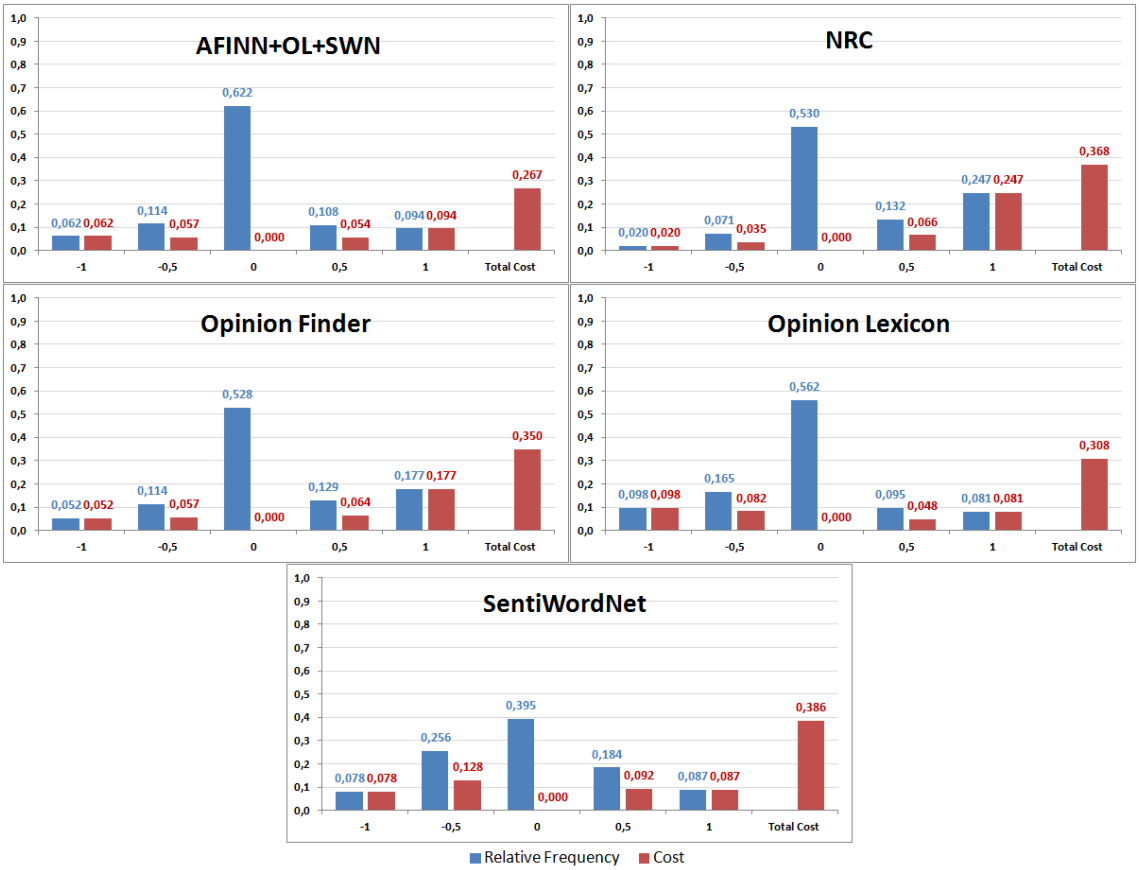

Figure B.5: Detailed cost analysis of the publicly available lexicons after lexicon enrichment.

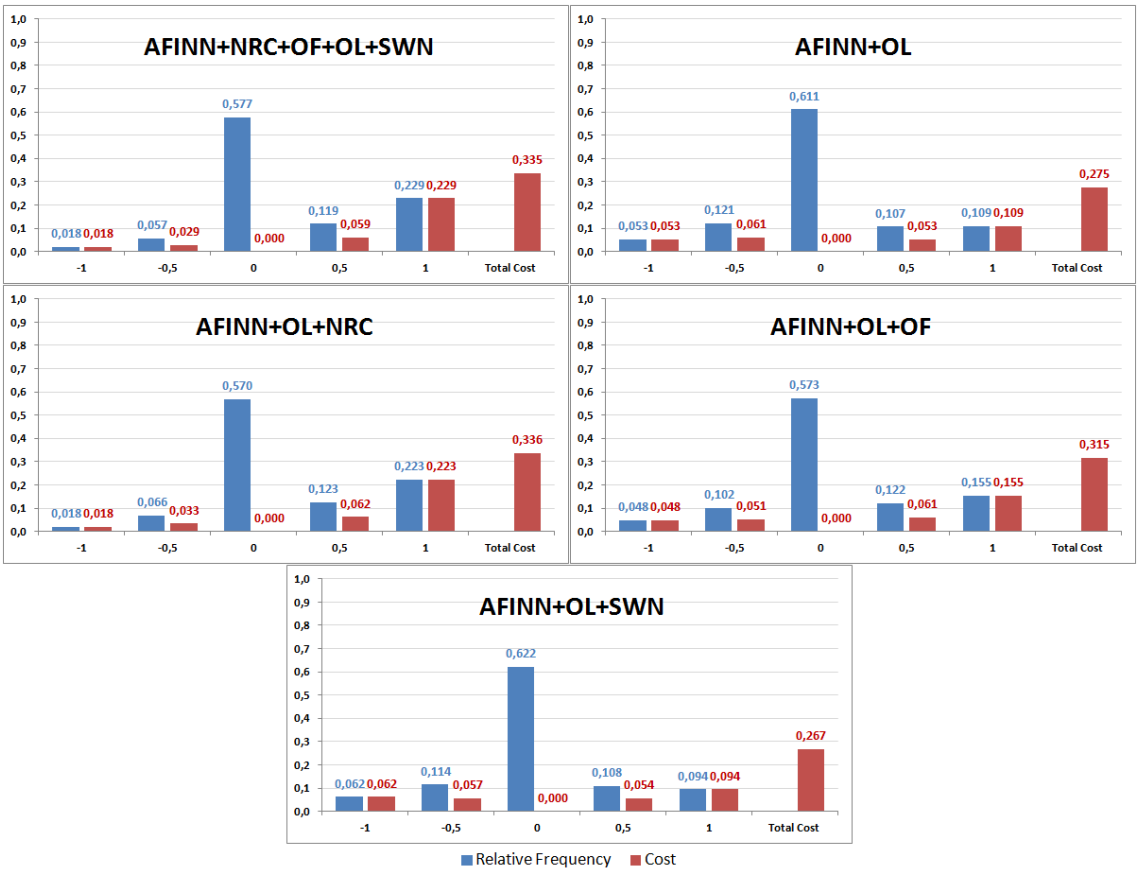

Figure B.6: Detailed cost analysis of the merged lexicons after lexicon enrichment.

Appendix C

Evaluation of Performance

C.1 Merging Lexicons

The evaluation results of the merge are available in Table C.1 and in Figure C.1.

				Negative class			Neutral class			Positive class		
No	Variant	*Micro F1*	*Macro F1*	*p*	*r*	*F1*	*p*	*r*	*F1*	*p*	*r*	*F1*
	AFINN	**0.570**	0.511	0.784	0.499	0.610	0.250	0.331	0.285	0.586	0.702	0.638
	NRC	0.521	0.401	0.713	0.268	0.390	0.212	0.138	0.167	0.524	0.847	0.647
	OF	0.509	0.441	0.700	0.429	0.532	0.187	0.223	0.204	0.528	0.663	0.588
	OL	0.553	0.514	0.671	0.676	0.673	0.253	0.423	0.317	0.626	0.492	0.551
	SWN	0.489	0.471	0.615	0.569	0.591	0.249	0.554	0.344	0.591	0.403	0.479
1	**AFINN+NRC+OF+OL+SWN**	**0.570**	0.431	0.762	0.364	0.493	0.229	0.085	0.124	0.547	0.888	0.677
2	**AFINN+OL**	**0.577**	0.502	0.731	0.577	0.645	0.216	0.231	0.223	0.595	0.685	0.637
3	AFINN+OL+AFFIN	0.552	0.421	0.759	0.350	0.480	0.171	0.092	0.120	0.542	0.859	0.,665
4	**AFINN+OL+NRC**	**0.570**	0.472	0.725	0.522	0.607	0.188	0.146	0.165	0.570	0.740	0.644
5	**AFINN+OL+SWN**	**0.589**	0.500	0.717	0.612	0.660	0.189	0.185	0.187	0.617	0.695	0.653

Table C.1: Classification results of merging different lexicons.

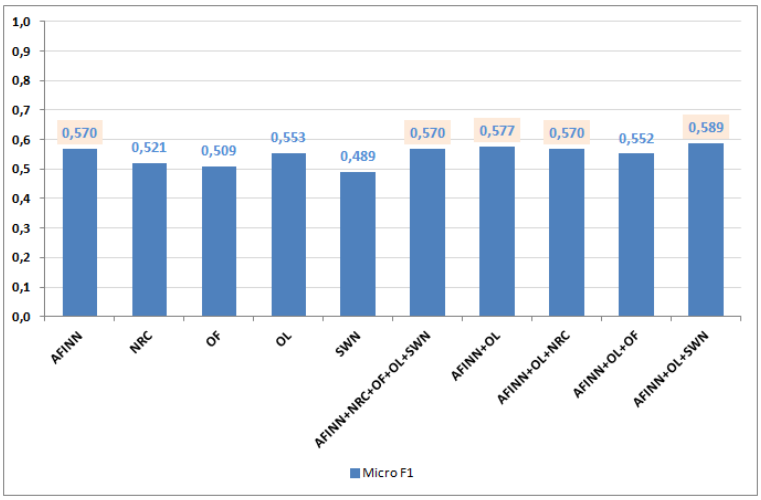

Figure C.1: *Micro F1* results after merging different lexicons.

By analysing Table C.1 and Figure C.1 it can be concluded that merging lexicons improves the classifier's performance, as better results were obtained when comparing with using AFINN alone (that had been considered the best classifier so far).

C.2 Negation Handling

The evaluation results after applying the negation technique are described below in Table C.2 and Figure C.2.

			Negative class			Neutral class			Positive class		
	Micro F1	*Macro F1*	*p*	*r*	*F1*	*p*	*r*	*F1*	*p*	*r*	*F1*
AFINN	0.557	0.503	0.781	0.490	0.602	0.246	0.338	0.285	0.572	0.680	0.622
NRC	0.517	0.393	0.684	0.265	0.382	0.180	0.123	0.146	0.528	0.845	0.650
OF	0.512	0.447	0.682	0.443	0.537	0.203	0.238	0.219	0.531	0.654	0.586
OL	0.547	0.508	0.669	0.665	0.667	0.245	0.415	0.309	0.622	0.492	0.549
SWN	0.500	0.478	0.602	0.638	0.620	0.259	0.546	0.351	0.614	0.372	0.464
AFINN+NRC+OF+OL+SWN	**0.581**	0.445	0.722	0.417	0.529	0.216	0.085	0.122	0.566	0.869	0.685
AFINN+OL	**0.578**	0.503	0.740	0.589	0.656	0.208	0.231	0.219	0.598	0.678	0.635
AFINN+OL+NRC	0.556	0.431	0.747	0.362	0.487	0.179	0.108	0.135	0.552	0.854	0.671
AFINN+OL+OF	0.570	0.475	0.717	0.510	0.596	0.200	0.162	0.179	0.575	0.745	0.649
AFINN+OL+SWN	**0.582**	0.494	0.679	0.659	0.669	0.189	0.177	0.183	0.618	0.644	0.631

Table C.2: Classification results after applying negation technique.

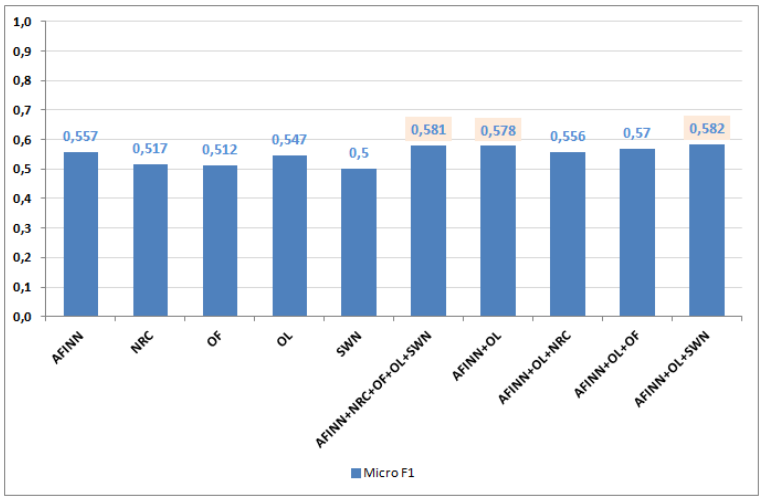

Figure C.2: *Micro F1* results after applying negation technique.

The merge of AFFIN, Opinion Lexicon and SentiWordNet continues to have the best performance, however its *Micro F1* decreased from 58.9% to 58.2%.

Table C.3 allows us to better compare if the results of applying this negation technique improved the sentiment classification.

	Micro F1 without applying negation	*Micro F1* after applying negation	Improvement
AFINN	**0.570**	0.557	-0.013
NRC	**0.521**	0.517	-0.004
OF	0.509	**0.512**	0.013
OL	**0.553**	0.547	-0.003
SWN	0.489	**0.500**	0.011
AFINN+NRC+OF+OL+SWN	0.570	**0.581**	0.011
AFINN+OL	0.577	**0.578**	0.001
AFINN+OL+NRC	0.552	**0.556**	0.004
AFINN+OL+OF	0.570	0.570	0.000
AFINN+OL+SWN	**0.589**	0.582	-0.007

Table C.3: Comparing *Micro F1* results before and after applying negation technique.

Negation handling improved the performance in some cases, but it had the opposite effect in others.

C.3 Lexicon Enrichment

The evaluation results of the classifiers after enriching the sentiment lexicons are shown in Table C.4 and Figure C.3.

			Negative class			Neutral class			Positive class		
	Micro F1	*Macro F1*	*p*	*r*	*F1*	*p*	*r*	*F1*	*p*	*r*	*F1*
AFINN	**0.586**	0.526	0.764	0.566	0.650	0.263	0.323	0.290	0.600	0.685	0.640
NRC	0.530	0.407	0.730	0.292	0.417	0.193	0.123	0.150	0.531	0.852	0.654
OF	0.528	0.452	0.709	0.455	0.554	0.187	0.200	0.193	0.542	0.690	0.607
OL	0.562	0.521	0.678	0.688	0.683	0.259	0.423	0.322	0.633	0.501	0.559
SWN	0.395	0.396	0.604	0.347	0.441	0.221	0.800	0.347	0.573	0.308	0.401
AFINN+NRC+OF+OL+SWN	0.577	0.438	0.774	0.379	0.509	0.224	0.085	0.123	0.554	0.893	0.684
AFINN+OL	**0.611**	0.533	0.760	0.656	0.704	0.234	0.246	0.240	0.627	0.687	0.656
AFINN+OL+NRC	0.570	0.440	0.778	0.379	0.510	0.200	0.100	0.133	0.553	0.871	0.677
AFINN+OL+OF	0.573	0.476	0.740	0.539	0.624	0.176	0.146	0.160	0.575	0.733	0.644
AFINN+OL+SWN	**0.622**	0.531	0.736	0.691	0.713	0.217	0.200	0.208	0.649	0.697	0.672

Table C.4: Classification results after lexicon enrichment.

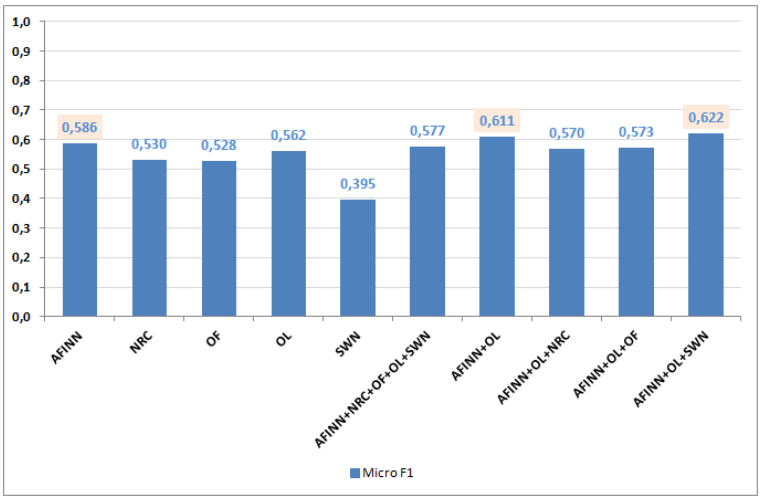

Figure C.3: *Micro F1* results after lexicon enrichment.

The following table compares the results before and after lexicon enrichment.

	Micro F1 without lexicon enrichment	*Micro F1* after lexicon enrichment	Improvement
AFINN	0.570	**0.586**	0.016
NRC	0.521	**0.530**	0.009
OF	0.509	**0.528**	0.019
OL	0.553	**0.562**	0.009
SWN	**0.489**	0.395	-0.094
AFINN+NRC+OF+OL+SWN	0.570	**0.577**	0.007
AFINN+OL	0.577	**0.611**	0.034
AFINN+OL+NRC	0.552	**0.570**	0.018
AFINN+OL+OF	0.570	**0.573**	0.003
AFINN+OL+SWN	0.589	**0.622**	0.033

Table C.5: Comparing *Micro F1* results before and after lexicon enrichment.

Analysing the Table C.5 results it can be concluded that:

- All lexicons, with the exception of SentiWordNet improved their performance.
- The lexicon that previously obtained the best result, AFINN+OL+SWN, with a *Micro F1* of 58.9%, got even better results with the incorporation of new terms in the lexicon, increasing *Micro F1* to 62.2%.
- Overall, developing a lexicon oriented to the subject studied was very beneficial to the sentiment classification task.

Appendix D

Lexicon Enrichment

In this section we present the words that were added or removed from lexicons in section 4.7.1.

D.1 Terms Removed From Lexicons

Negative words removed manually from lexicons
ax

Table D.1: List of negative words manually removed from lexicons.

Positive words removed manually from lexicons
diverting
gold
indebted
influenza
share
shares
worth

Table D.2: List of positive words manually removed from lexicons.

D.2 Terms Added to Lexicons

Negative words added manually to lexicons
awash
bailouts
belt-tightening
bottleneck
bottlenecks
cash-strap (stem of cash-strapped)
chemotherapy
crimea
damag (stem of damage)
declined
diverted
diverting
down
expenses
fines
forcing
indebted
influenza
ipo
low-income
opposit (stem of opposition)
overshadowing
punishment
radiation
recused
sidestepped
smaller-than-expected
subprime
takeov (stem of takeover)
takeover
takeovers
uncertainti (stem of uncertainty)
vanish
vanished
vanishing
wars
withdrawals
wreckage

Table D.3: List of negative words manually added to lexicons.

Positive words added manually to lexicons
agreements
better-than-expected
biofuel
cancer-free
confid (stem of confident)
consolidation
curable
dividend
earned
funded
funding
incremental
invested
investing
revamp
revamped
self-sustaining
settle
settled
settling
up

Table D.4: List of positive words manually added to lexicons.