

2009-03-01

# Opinion mining with the SentWordNet lexical resource

Bruno Ohana

*Technological University Dublin*, bohana@gmail.com

# Opinion Mining with the SentiWordNet Lexical Resource

**Bruno Ohana**

A dissertation submitted in partial fulfilment of the requirements of
Dublin Institute of Technology for the degree of
M.Sc. in Computing (Knowledge Management)

**March 2009**

I certify that this dissertation which I now submit for examination for the award of MSc in Computing (Knowledge Management), is entirely my own work and has not been taken from the work of others save and to the extent that such work has been cited and acknowledged within the test of my work.

This dissertation was prepared according to the regulations for postgraduate study of the Dublin Institute of Technology and has not been submitted in whole or part for an award in any other Institute or University.

The work reported on in this dissertation conforms to the principles and requirements of the Institute's guidelines for ethics in research.

***Signed:*** _____

***Date:*** ***31 March 2009***

# 1. ABSTRACT

Sentiment classification concerns the application of automatic methods for predicting the orientation of sentiment present on text documents. It is an important subject in opinion mining research, with applications on a number of areas including recommender and advertising systems, customer intelligence and information retrieval.

SentiWordNet is a lexical resource of sentiment information for terms in the English language designed to assist in opinion mining tasks, where each term is associated with numerical scores for positive and negative sentiment information. A resource that makes term level sentiment information readily available could be of use in building more effective sentiment classification methods.

This research presents the results of an experiment that applied the SentiWordNet lexical resource to the problem of automatic sentiment classification of film reviews. First, a data set of relevant features extracted from text documents using SentiWordNet was designed and implemented. The resulting feature set is then used as input for training a support vector machine classifier for predicting the sentiment orientation of the underlying film review. Several scenarios exploring variations on the parameters that generate the data set, outlier removal and feature selection were executed.

The results obtained are compared to other methods documented in the literature. It was found that they are in line with other experiments that propose similar approaches and use the same data set of film reviews, indicating SentiWordNet could become an important resource for the task of sentiment classification. Considerations on future improvements are also presented based on a detailed analysis of classification results.

**Key words:** opinion mining, sentiment classification, lexical resources, data mining, SentiWordNet.

To my Parents

# ACKNOWLEDGEMENTS

I would like to express my sincere thanks to my supervisor, Brendan Tierney, for all the help and guidance provided on all stages of this dissertation and for the valuable insights and discussions that contributed to the results of this project.

I also wish to thank Andrea Esuli and Fabrizio Sebastiani, from the Italian Institute of Information Science and Technology, for promptly making the SentiWordNet lexical resource available for use in this research.

# TABLE OF CONTENTS

**1. ABSTRACT ..... II**

**TABLE OF FIGURES ..... IX**

**TABLE OF TABLES ..... X**

**1. INTRODUCTION ..... 12**

1.1. KNOWLEDGE MANAGEMENT, DISCOVERY AND OPINIONS IN TEXT ..... 12
1.2. BACKGROUND ..... 14
1.3. RESEARCH PROBLEM ..... 16
1.4. THE INTELLECTUAL CHALLENGE ..... 16
1.5. RESEARCH OBJECTIVES ..... 17
1.6. METHODOLOGY ..... 18
1.7. RESOURCES ..... 19
1.8. SCOPE AND LIMITATIONS ..... 20
1.9. ORGANISATION OF THIS DISSERTATION ..... 21

**2. KNOWLEDGE CREATION AND DISCOVERY ..... 23**

2.1. KNOWLEDGE ORGANISATIONS ..... 23
2.2. KNOWLEDGE MANAGEMENT ..... 25
*2.2.1. Characterising Knowledge ..... 27*
2.3. KNOWLEDGE CREATION ..... 31
*2.3.1. Nonaka's Spiral of Knowledge (Nonaka, 1994) ..... 31*
*2.3.2. The Conditions for Knowledge Creation ..... 33*
2.4. INFORMATION TECHNOLOGY AND KNOWLEDGE MANAGEMENT ..... 35
2.5. KNOWLEDGE DISCOVERY ..... 38
*2.5.1. Knowledge Discovery and Data Mining ..... 39*
*2.5.2. Implications to Knowledge Management ..... 40*
2.6. CONCLUSION ..... 41

**3. KNOWLEGDE DISCOVERY AND DATA MINING ..... 43**

3.1. KNOWLEDGE DISCOVERY PROCESSES ..... 43
*3.1.1. Analysis of Knowledge Discovery Processes ..... 44*

*3.1.2. Considerations on Knowledge Discovery Processes* ..... *48*
3.2. DATA MINING TECHNIQUES ..... 52
*3.2.1. Goal Based Categorisation of Data Mining Techniques* ..... *52*
*3.2.2. Summary and Considerations* ..... *57*
*3.2.3. Considerations on the Data Set* ..... *58*
3.3. DATA MINING ALGORITHMS FOR CLASSIFICATION ..... 61
*3.3.1. Introduction* ..... *61*
*3.3.2. Supervised Learning Algorithms* ..... *62*
*3.3.3. Nearest Neighbour Methods* ..... *63*
*3.3.4. Tree Based Methods* ..... *65*
*3.3.5. Naïve Bayes* ..... *66*
*3.3.6. Large Margin Classifiers: Support Vector Machines* ..... *67*
*3.3.7. Considerations on Classifier Techniques* ..... *69*
*3.3.8. Evaluating Classifier Performance* ..... *72*
*3.3.9. Challenges to Classification in Data Mining* ..... *75*
3.4. DATA MINING TOOLS ..... 79
*3.4.1. Open Source Tools and RapidMiner* ..... *80*
3.5. CONCLUSION ..... 81

**4. TEXT MINING AND OPINION MINING ..... 83**

4.1. TEXT MINING ..... 83
*4.1.1. Applications of Text Mining* ..... *86*
*4.1.2. Representation of Text Data* ..... *88*
*4.1.3. Document Classification Techniques* ..... *93*
4.2. OPINION MINING ..... 95
*4.2.1. Introduction: Opinions in Text* ..... *95*
*4.2.2. Key Problems Addressed by Opinion Mining Research* ..... *98*
*4.2.3. Subjectivity Detection* ..... *100*
*4.2.4. Sentiment Classification* ..... *101*
*4.2.5. Lexical Resources for Opinion Mining and SentiWordNet* ..... *106*
4.3. CONCLUSION ..... 111

**5. DESIGNING FEATURES WITH SENTIWORDNET ..... 113**

5.1. INTRODUCTION ..... 113

5.2. THE SENTIWORDNET DATABASE ..... 113
*5.2.1. Database Structure ..... 114*
*5.2.2. Statistics on Part of Speech Scoring ..... 115*
5.3. CONSIDERATIONS ON SENTIWORDNET DATA ..... 116
*5.3.1. Automatic Part of Speech Tagging ..... 116*
*5.3.2. Word Sense Disambiguation ..... 118*
5.4. THE POLARITY DATA SET ..... 119
*5.4.1. Document Structure ..... 120*
*5.4.2. Considerations on Writing Style ..... 121*
5.5. PROPOSED MODEL ..... 124
5.6. CONCLUSION ..... 129
**6. SENTIMENT CLASSIFICATION EXPERIMENT ..... 131**
6.1. INTRODUCTION ..... 131
6.2. OBJECTIVES AND SCOPE ..... 132
*6.2.1. Out of Scope ..... 134*
6.3. EXPERIMENT PROCESS ..... 134
*6.3.1. Baseline Classifier ..... 135*
*6.3.2. Generate SentiWordNet Features ..... 136*
*6.3.3. Sentiment Classification Using SentiWordNet ..... 136*
6.4. CRITERIA FOR COMPARISONS ..... 138
*6.4.1. Classification Performance ..... 139*
*6.4.2. Training Data Set Size ..... 139*
*6.4.3. Dimensionality and Runtime ..... 139*
6.5. EXPERIMENT SETUP ..... 140
*6.5.1. Technical Resources ..... 140*
*6.5.2. Baseline Classifier ..... 141*
*6.5.3. SentiWordNet Test Approach ..... 142*
6.6. CONCLUSION ..... 147
**7. EXPERIMENT RESULTS ..... 148**
7.1. INTRODUCTION ..... 148
7.2. SENTIMENT CLASSIFICATION RESULTS ..... 148
*7.2.1. Baseline Results ..... 149*

*7.2.2. SentiWordNet Parameter Tests ..... 150*
*7.2.3. Outlier and Feature Selection ..... 154*
*7.2.4. Training Data Set Size and Execution Time ..... 157*
7.3. RESULT ANALYSIS AND CONSIDERATIONS ..... 158
*7.3.1. Accuracy Results ..... 159*
*7.3.2. Baseline Comparisons ..... 161*
*7.3.3. Analysis of Misclassifications ..... 166*
7.4. CONCLUSION ..... 170

**8. CONCLUSION ..... 174**

8.1. INTRODUCTION ..... 174
8.2. RESEARCH OVERVIEW AND OBJECTIVES ..... 175
8.3. EXPERIMENT RESULTS ..... 176
8.4. ADDITIONS TO THE BODY OF KNOWLEDGE ..... 178
8.5. FUTURE WORK & RESEARCH ..... 179
*8.5.1. SentiWordNet Features ..... 179*
*8.5.2. Classification Results ..... 180*
*8.5.3. Knowledge Management Research ..... 181*
8.6. CONCLUSIONS ..... 181
*8.6.1. Final Remarks ..... 182*

**REFERENCES ..... 183**

**APPENDIX A – LANGUAGE RESOURCES ..... 203**

A.1 PENN TREEBANK TAGSET ..... 203
A.1 STOP WORD LIST ..... 205

**APPENDIX B – PYTHON CODE ..... 206**

B.1 NEGATION ALGORTIHM ..... 206

# TABLE OF FIGURES

FIGURE 1 - ORGANISATION OF DISSERTATION CHAPTERS ..... 22
FIGURE 2 - THE KNOWLEDGE PYRAMID (AWAD ET AL, 2004) ..... 30
FIGURE 3 - STAGES OF THE KDD PROCESS (FAYYAD ET AL, 1996) ..... 44
FIGURE 4 - REVISED KDD PROCESS BY (COLLIER ET AL, 1998) ..... 46
FIGURE 5 - STAGES IN THE CRISP-DM METHODOLOGY (CHAPMAN ET AL, 2000) ..... 48
FIGURE 6 - CYCLES OF KNOWLEDGE DEVELOPMENT THROUGH DATA MINING (WANG ET AL, 2008) ..... 50
FIGURE 7 - EXAMPLE CLASSIFICATION OF LOAN APPLICATIONS ..... 54
FIGURE 8 - HYPERPLANE SEPARATING TWO CLASSES ..... 67
FIGURE 9 - RAPIDMINER WORKBENCH ..... 81
FIGURE 10 – EXAMPLE WORD VECTOR ..... 92
FIGURE 11 - SENTIWORDNET SAMPLE SCORE (HTTP://SENTIWORDNET.ISTI.CNR.IT) ... 108
FIGURE 12 - EXPERIMENT EXECUTION STAGES ..... 135
FIGURE 13 - SENTIWORDNET SENTIMENT CLASSIFICATION EXPERIMENT ..... 143
FIGURE 14 - ACCURACY COMPARISONS FOR VARIOUS CROSS-VALIDATION FOLDS .... 162
FIGURE 15 - 3-FOLD ACCURACIES FOR DIFFERENT TRAINING SET SIZES ..... 163

# TABLE OF TABLES

TABLE 1 - KM LIFE CYCLE (AWAD ET AL, 2004) ..... 27
TABLE 2 - MODES OF KNOWLEDGE CREATION (NONAKA, 1994)..... 32
TABLE 3 - INFORMATION SYSTEMS AND KNOWLEDGE CONVERSIONS (MARWICK, 2001) ..... 35
TABLE 4 - DATA MINING TECHNIQUES CATEGORISED BY GOAL ..... 58
TABLE 5 - EXAMPLE DATA SET FOR LOAN APPLICATIONS..... 59
TABLE 6 - SURVEY OF CLASSIFICATION METHODS ..... 70
TABLE 7 - CONFUSION MATRIX FOR 2-CLASS CLASSIFICATION PROBLEM ..... 73
TABLE 8 - KEY ISSUES IN TEXT MINING (STAVRIANOU, 2007) ..... 89
TABLE 9- SENTIWORDNET DATABASE RECORD STRUCTURE..... 115
TABLE 10 - SAMPLE SENTIWORDNET DATA ..... 115
TABLE 11 - SCORING STATISTICS PER PART OF SPEECH (ESULI ET AL, 2006)..... 116
TABLE 12 - PENN TREEBANK TAGS (MARCUS ET AL, 1993) FOR PARTS OF SPEECH PRESENT IN SENTIWORDNET ..... 117
TABLE 13 - EXAMPLE OF MULTIPLE SCORES FOR THE SAME TERM IN SENTIWORDNET 118
TABLE 14- POLARITY DATA SET DOCUMENT STATISTICS ..... 121
TABLE 15 - PROCESS DIAGRAM FOR GENERATING SENTIWORDNET FEATURES ..... 125
TABLE 16 - SENTIWORDNET FEATURE DESCRIPTION..... 128
TABLE 17 - PARAMETERS FOR FEATURE GENERATION ..... 129
TABLE 18 - SOFTWARE AND HARDWARE RESOURCES ..... 141
TABLE 19 -CLASSIFIERS AND PARAMETERS TESTED IN EXPERIMENT ..... 144
TABLE 20 - DEFAULT SENTIWORDNET PARAMETERS ..... 144
TABLE 21 - PARAMETER VALUES TESTED BY EXPERIMENT..... 145
TABLE 22 - BASELINE RESULTS AND WORD VECTOR TYPES ..... 149
TABLE 23 - RESULTS FOR BASELINE CLASSIFIER USING BINARY PRESENCE WORD VECTOR ..... 149
TABLE 24 - RESULTS FOR SENTIWORDNET FEATURES USING 3 CLASSIFICATION ALGORITHMS ..... 150
TABLE 25 - ACCURACY RESULTS FOR VARYING SCORING FUNCTIONS ..... 151
TABLE 26 - ACCURACY RESULTS FOR VARYING SCORING THRESHOLD VALUES ..... 152

TABLE 27 - ACCURACY RESULTS FOR NEGATION ALGORITHM WITH VARYING WINDOW SIZES ..... 153
TABLE 28 - ACCURACY RESULTS FOR VARYING NUMBER OF SEGMENTS ..... 153
TABLE 29 - BEST SENTIWORDNET PARAMETERS OBTAINED WITH EXPERIMENT ..... 154
TABLE 30 - ACCURACY RESULTS WITH OUTLIER REMOVAL ..... 155
TABLE 31 - 20 SENTIWORDNET FEATURES WITH LOWEST CORRELATION TO LABEL USING CHI-SQUARED TEST ..... 156
TABLE 32 - ACCURACY RESULTS WITH FEATURE REMOVAL ..... 156
TABLE 33 -ACCURACY AND EXPERIMENT TIMINGS WITH OUTLIER DETECTION ..... 157
TABLE 34 - ACCURACY AND EXPERIMENT TIMINGS WITHOUT OUTLIER DETECTION .. 158
TABLE 35 - ACCURACIES FOR VARIOUS TRAINING SET SIZES ..... 158
TABLE 36 - SENTIWORDNET SENTIMENT CLASSIFICATION RESULTS ..... 159
TABLE 37 - SENTIWORDNET AND BASELINE CLASSIFICATION ACCURACY ..... 162
TABLE 38 - EXECUTION TIMES FOR BASELINE AND SENTIWORDNET CLASSIFIERS ..... 164
TABLE 39 - TRAINING TIMES FOR BASELINE AND SENTIWORDNET CLASSIFIERS ..... 165
TABLE 40 - ACCURACY COMPARISON WITH PUBLISHED RESEARCH ..... 176

# 1. INTRODUCTION

*"Where is the wisdom that we have lost in knowledge?*
*Where is the knowledge that we have lost in information?"*
*T.S. Eliot.*

## *1.1. Knowledge Management, Discovery and Opinions in Text*

In the recent decades, the business world has witnessed a number of changes that affected the competitive landscape of companies across all industries. The impressive advances in technology, deregulation trends and the lowering of trade barriers resulted in accelerated competition and the need for companies to constantly innovate its products and services. Those changes have influenced how a company should be positioned, and what strategies should companies pursue to achieve or preserve their competitive advantage. From a transactional view of the company, whose *raison d'etre* was to process a certain input into a value added output, emerged the notion of a knowledge creating company, capable of constantly innovate and improve its products, business processes and services. Naturally, one key element of the innovation processes behind this dynamic nature is the knowledge that exists within the organisation in the form of people's skills and experiences, or stored in databases and other repositories. Ensuring this knowledge is efficiently managed, and effectively employed to maximise the success of organisations is the realm of the discipline of *Knowledge Management*.

At the same time the huge advances in information technology, coupled with the reduction in cost of the technology infrastructure has caused a true explosion in the volumes of data available in information systems. Today, most aspects of an organisation's business processes are dependent on computer systems such as online collaboration, email, transactional databases and data warehouses. The volumes of information are indeed much larger than an individual's ability to process them, causing a phenomenon whereby too much information is leading to inefficiencies in decision making: information overload (Farhoomand et al, 2002). From a knowledge management standpoint, the inability to tap into the vast information resources stored on computer systems also affects a company's capacity to reuse existing knowledge

already created, and to create new knowledge from yet undiscovered patterns and relationships present in data. Providing means for creating new knowledge is an important consideration when shaping the innovation strategy in organisations, and knowledge discovery from data repositories can be an important factor on such strategies (Wang et al, 2008).

The ability to discover new knowledge from databases using automated methods thus became relevant to the success of organisations, and a requirement for the detailed analysis of any very large set of data. This need has fuelled research in the area of knowledge discovery in databases, or data mining, and the development of methodologies, techniques and systems to execute this type of project. Meanwhile a strong industry has also been developed, dedicated to assisting organisations in their knowledge discovery initiatives, and applications of data mining techniques have become relevant in a number of real world scenarios such as recommendation systems, spam filtering, customer trend analysis, and fraud.

One important type of information available in computer systems today is textual data. This is by far the most widely used method for storing information in explicit form, and some estimates suggest that as far as 85% of information found in organisations is in text format (McKnight, 2005). It is also the most common source of information on the internet, being the natural way of presenting information in human readable form. It is in this context that performing data mining on text or, *Text Mining* gains importance. Text mining applies knowledge discovery methods to unstructured textual data, leveraging other research areas such as natural language processing, artificial intelligence and machine learning to tackle the complexities of extracting information from unstructured textual format. Text mining techniques have been applied in a number of knowledge discovery scenarios, such as automatic categorisation of documents, trend analysis and spam detection.

An important branch of research within knowledge discovery in text concerns the ability to detect and extract opinions, or sentiment information. Detecting the sentiment of customers towards a new product based on feedback available in text format could be an important element affecting decision making and the product's

future direction. *Opinion Mining* is the research area dealing with automated methods for detecting and extracting this information from textual data, and has a number of potential applications on building more efficient recommender systems, financial analysis, product engineering and market research. One approach for detecting sentiment in text present in literature proposes the use of sentiment-oriented lexical resources such as a dictionary of opinionated terms. *SentiWordNet* is one such lexical resource containing opinion bias information on terms extracted from the WordNet database, and was made publicly available for research purposes (Esuli et al, 2006). Lexical resources have the advantage of being created a-priori and the potential of being applied to a number of different categories of text. This, an interesting question is to assess how effective are lexical resources to detecting sentiment, in comparison to other methods, and the potential advantages that could be obtained from this approach.

## *1.2. Background*

Within opinion mining research, sentiment classification concerns the application of automatic methods for making predictions about the *orientation* of sentiment present on text documents. These predictions are given according to pre-defined values for sentiment polarity. For instance, the sentiment of film reviews could be classified as positive, or "thumbs-up", or negative "thumbs-down"; author sentiment on articles of a given subject, such as a proposed tax bill could be subject to similar types of queries, and ranked in a numeric scale representing sentiment strength and orientation.

Sentiment classification is a valuable technology in a number of fields such as recommender systems that can retrieve information based on sentiment orientation, as an aide to the correct placement of online advertising by evaluating the sentiment of a page's content; or in online collaboration systems where sentiment detection can assist the detection of inappropriate user behaviour, or *flaming*. The problem of evaluating sentiment orientation for the purposes of classification has received considerable research attention, and several approaches are surveyed in (Pang et al, 2008). One of the seminal experiments published in the literature is reported in (Pang et al, 2002), where well known bag-of-words machine learning methods used in text classification were applied to sentiment classification using a data set of film reviews. The data set used for the experiments is known as the *polarity data set*, comprising 2000 film

reviews extracted from discussion groups from the internet movie database, and was also made available for further research in (Pang et al, 2004).

It was observed that the results obtained using text classification methods based on the bag-of-words approach seen in (Pang et al, 2002) remained below that of traditional topic based text classification, suggesting the extraction of patterns that capture sentiment information in text requires additional linguistic analysis, and has fuelled interest in this field of research. In (Cui et al, 2006) an experiment indicates that the use of higher order n-grams based on pairs of words and three word combinations can yield better classification results, provided the training data set is sufficiently large. Another approach suggesting the use of linguistic part of speech information as features is seen in (Salvetti et al, 2004) and (Wiebe et al, 2003). The relationship between sentiment orientation and the detection of subjective and objective sentences within a document is explored in (Pang et al, 2004), with considerable improvements over the baseline bag-of-words method.

Finally, it can be argued that sentiment information exists at term level through words and expressions known to carry a given sentiment polarity. Intuitively, a product review that contains words such as “excellent” and “good” can be expected to be more likely a positive than a negative review. There are several methods that explore the existence of such words and perform sentiment classification based on calculating scores based on terms present in a document from pre-defined lists of positive and negative terms. Examples of opinion mining experiments implementing techniques based on term sentiment as seen in (Salvetti et al, 2004), (Pang et al, 2002) and (Kennedy et al, 2006).

SentiWordNet is a lexical resource of sentiment information for terms in the English language introduced in (Esuli et al, 2006) designed to assist in opinion mining tasks. Each term in SentiWordNet is associated with numerical scores for positive and negative sentiment information. The database is built upon a subset of paradigmatic terms assumed a priori carry positive or negative sentiment, such as the words “good” and “bad”, and extended by an iterative process that generates scores based on term relationships extracted from the WordNet database (Miller et al, 1990). Investigating the potential benefits of using the SentiWordNet database for performing sentiment

classification is the key purpose of this dissertation's research, and is explored further in the next section.

## *1.3. Research Problem*

SentiWordNet could be a valuable resource for performing opinion mining tasks since it provides a readily available database of term sentiment information for the English language. This means SentiWordNet can be a replacement to the process of manually deriving lists of terms containing sentiment information for opinion mining tasks. It can also be noted that SentiWordNet is built from a semi automated process that derives opinion information from the WordNet database, and has the potential to be applied to documents on different domains. The semi-automated approach also indicates the process can easily be replicated on other languages, where lexicons similar to WordNet are available.

Thus, SentiWordNet offers potential benefits to opinion mining and to the task of sentiment classification in particular. Assessing the viability and performance of SentiWordNet as a tool for performing sentiment classification on textual documents is the key research problem of this dissertation, and the results can provide useful insights on its application to opinion mining tasks, and further research direction for this type of lexical resource.

## *1.4. The Intellectual Challenge*

To tackle the research problem proposed above, the main challenges of this dissertation are related firstly to the design of a set of features extracted in conjunction with SentiWordNet that capture as much sentiment information as possible from text documents. The feature design was based on a detailed evaluation of the SentiWordNet database, the data set chosen for the experiment, a study of other approaches proposed previously in opinion mining research, and finally in identifying and understanding the limitations of sentiment classification based on term information.

Secondly, designing and executing an experiment that leverages data mining techniques to perform sentiment classification with SentiWordNet required

understanding of the state of the art in data mining algorithms for classification, and how they relate to previous efforts in sentiment classification reported in the literature. Also, the ability to implement the proposed experiment in a data mining package, and building the necessary technical components to extract SentiWordNet information are also important challenges this research faced.

## *1.5. Research Objectives*

Having in mind the research problem and intellectual challenges posed in the previous sections, the objectives of this dissertation's research can be outlined as follows:

- Investigate the fields of knowledge management, knowledge discovery and data mining, and the relevance of data mining for the creation of new knowledge and organisation competitiveness.

- Review research in data mining processes, the state of the art in algorithms for classification, challenges and limitations to classifier performance; investigate the state of the art in data mining tools.

- Investigate the areas of text mining and opinion mining, outline approaches proposed in the literature for performing sentiment classification; review literature on lexical resources used in opinion mining and SentiWordNet.

- Design a data set with features extracted with the help of SentiWordNet, to be used for sentiment classification of text documents.

- Implement an algorithm that extracts the proposed features using SentiWordNet, and using the polarity data set (Pang et al, 2002) as the source for text documents.

- Design and train a baseline classifier for sentiment classification similar to the one presented in (Pang et al, 2002), to be used for comparisons.

- Design and train a classifier based on SentiWordNet features for sentiment classification.

- Evaluate the effects of changing generation parameters of SentiWordNet features to the overall performance of sentiment classification.

- Evaluate the effects of outlier removal and feature selection to overall performance of sentiment classification using SentiWordNet.

- Analyse results obtained, investigate source of classification errors, and compare results with other research in the literature using the same data set.

## *1.6. Methodology*

As part of this dissertation, both primary and secondary research was conducted. Secondary research consisted of a review on research literature on the fields of knowledge management, knowledge discovery, data mining, text mining and opinion mining. The following resources were used to perform secondary research.

- Research journals and periodicals (ACM, IEEE, Harvard Business Review, etc.).
- Published books in the relevant areas.
- Conference Proceedings.
- Websites and discussion groups associated to relevant research.
- Product white papers.
- Company websites.

The primary research is an experiment in sentiment classification that uses data mining techniques and SentiWordNet. The methodology is based on best practices found on knowledge discovery methodologies from both the industry and academic circles, involving an iterative process of:

- Data selection and data pre-processing tasks originating from a data set of film reviews in raw text format.

- Implementation of sentiment classification experiments using machine learning algorithms, applying to this end a data mining application that allows for the rapid prototyping of tasks.

- Results evaluation and comparisons.

The experiment setup and presentation of results will closely follow published work in the area, to ensure result comparisons are possible, and that the experiment follows sound research practices.

## *1.7. Resources*

To successfully achieve the goals of this dissertation, the following resources were identified as key requirements.

**Human Resources**

- Access to supervisor, for review and guidance throughout the preparation of the dissertation.

- Access to other members of DIT research staff as needed, for addressing more technical questions and sharing ideas.

**Technical Resources**

- Personal Computer system or laptop of recent specification for setting up and executing experiment.

- Access to library resources for research in books and periodicals.
    - Due to the nature of a part-time degree, online and remote access to content should be used whenever possible.

- Data Mining Application.

    - The *RapidMiner* open source tool will be employed (Mierswa et al, 2006).

- Programming language.
    - Python programming language and libraries will be used for required programming tasks during experiment execution.
    - The Python NLTK Library will also be employed for specific linguistic tasks (Loper et al, 2002).

- Labelled text mining data sets: The *polarity* data set (Pang et al, 2004) comprising of text corpora of film reviews will be used for the experiment.

- SentiWordNet Database (Esuli et al, 2006)
    - The lexical resource is required in a format that can easily be integrated into the experiment.

## *1.8. Scope and Limitations*

The experiment conducted on this research uses the well known polarity data set for execution and presentation of results. This is useful for comparisons to other research in opinion mining; however applying this research to other data sets could yield different results and new insights. It is acknowledged that testing on a single data set is a limitation of this research.

The main focus of this research is evaluating sentiment classification using SentiWordNet and comparing it to other approaches in the literature. To this end, choosing classification algorithms and algorithm parameters are a pre-requisite step for the data mining aspect of the experiment. Whereas potentially better results are possible by using a different choice of parameter than the ones presented here, it is not the objective of this research to investigate this aspect of data mining. Instead, to stay within the focus of the experiment, the choice of parameters and algorithms will be based on previous results in the literature, and a limited evaluation by experimentation.

## *1.9. Organisation of this dissertation*

The remaining chapters of this dissertation are organised into the review of relevant research, experiment design and execution, and results presentation and conclusion according to the chapter described below.

**Chapter 2** presents a review of research literature in knowledge management and knowledge discovery, stressing the importance of knowledge creation as a knowledge management initiative beneficial to the success of organisations, and the relationship between knowledge creation and the discovery of new knowledge stored in databases across various information systems.

**Chapter 3** reviews in more details the fields of knowledge discovery and data mining, evaluating methodologies for implementing knowledge discovery projects and the important success factors. The main data mining activities are reviewed, presenting a number of potential uses to data mining techniques, followed by a detailed analysis of pattern classification, algorithms and challenges. The chapter concludes with a review of data mining tools available today for commercial and academic use.

**Chapter 4** introduces the area of text mining and opinion mining, exploring their additional challenges and how they relate to the general area of data mining. Opinion Mining is the main subject of this dissertation and its research literature is reviewed in depth. The SentiWordNet lexical resource is presented and potential applications of such a resource are discussed.

In **Chapter 5** the first part of this dissertation's experiment is presented: an approach for extracting sentiment information from text documents as features for sentiment classification using SentiWordNet is proposed, based on a detailed evaluation of this lexical resource, and considerations on the challenges for extracting information from textual data.

**Chapter 6** describes the opinion mining experiment: the experiment's scope and objectives are laid out, the setup of the experiment, success criteria and limitations are discussed.

**Chapter 7** presents the experiment results. The results for each experiment activity proposed in Chapter 6 are presented as collected in accordance with the proposed metrics. The obtained results are examined in more details and discussed in light of other research in the literature.

Finally, **Chapter 8** concludes this dissertation. It reviews the dissertation's key objectives, the research approach and results obtained. The key contributions to the body of knowledge resulting of this research are presented, along with opportunities for future research. The chapter concludes with final remarks on the overall dissertation project.

The below diagram illustrates the division of Chapters according to its key objectives.

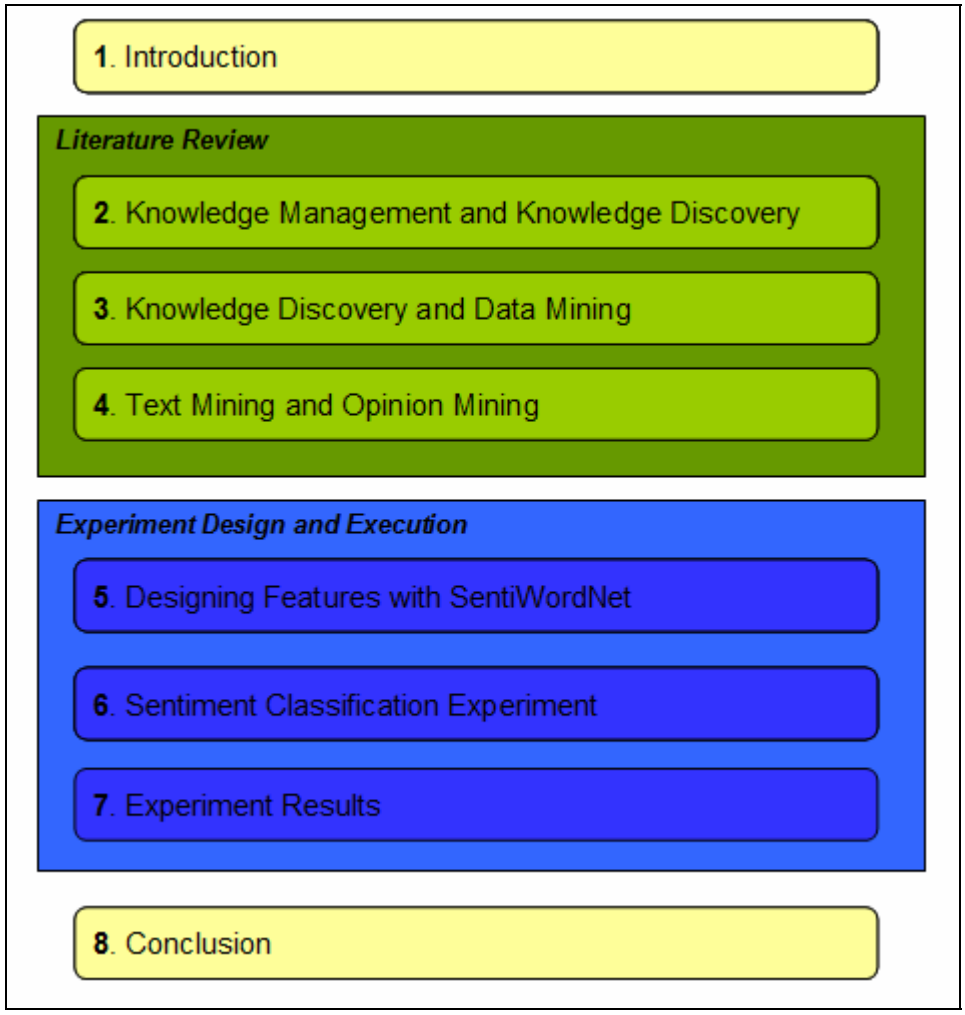



**Figure 1 - Organisation of Dissertation Chapters**

# 2. KNOWLEDGE CREATION AND DISCOVERY

In this chapter, research literature on the fields of knowledge management and knowledge discovery are presented and discussed. The discussion focuses on the importance of knowledge management and knowledge creation in particular as strategic tools for promoting organisational competitiveness, and how the vast amounts of data stored in companies' information systems can be used as a source for creating new knowledge via the process of knowledge discovery.

## *2.1. Knowledge Organisations*

The term *knowledge organisation* derives from understanding of the concept of a company from a new perspective that gained popularity in the strategic management literature through the 1990s (Cole, 1998; Alavi and Leidner, 2001; Nonaka, 1995). This view evolved from the traditional notion of the company as an information processing unit aimed at employing its resources to build a product or solve a specific problem. Knowledge resources have always been employed by organisations in the development and production of tangible goods, and the management of enterprises. It has been available, for example in the form of books, technical manuals, training and company communications. However the "transactional" view of the company did not take into account the fact that knowledge is not only applied but also dynamically created, exchanged and refined within the boundaries of the company, and that such phenomena influence the company's ability to grow, innovate and remain competitive (Nonaka, 1994).

The knowledge based view of the company gained momentum amid changes in competitive pressures faced by organisations. These changes precipitated in the need for new perspectives in the frameworks of how a company should be interpreted and analysed to ensure sustainable profitable positions in their industries:

- *Globalization and Deregulation* have in recent decades removed old trade barriers and expanded markets. Bigger markets also meant a wider number of competitors, and the lowering of trade barriers would over time lessen the impact of geographical advantages a company might have obtained from its location.

- *Intellectual Property* became increasingly regulated, thus protecting company's investment on knowledge assets by means of copyright laws, trademarks and patents (Teece, 1998). This provided a stable framework that favoured the creation and growth of companies whose business is to commercialise those assets.

- In addition, the idea of *The Law of Increasing Returns* have influenced economic understanding of companies, stating that knowledge based activities are not subject to the traditional model of diminishing returns, where companies' return on investment tend to reduce over time in the face of increased competition and marginal cost increases. Instead, knowledge based activities are more difficult to imitate, and tend to create a positive feedback loop that amplifies returns as more knowledge gets created from already existing knowledge. This aggregated with factors such as customer's technology lock-ins and appropriate market timing suggest increasing potential returns of knowledge assets over time (Teece, 1998; Arthur, 1996).

These factors have strengthened the idea of knowledge as the source of sustained creation of new products and services, and the ultimate resource for increasing company value and its competitiveness (Cole, 1998; Nonaka, 1991). In the face of increased competition in a global market, and promising prospects that can be achieved from developing knowledge assets, it became more suitable to view the firm as a collection of capabilities and knowledge skills that can be quickly reconfigured and applied in other realms, a notion later crystallised in the term knowledge organisation (Davenport, 2001; Nonaka, 1994; Drucker, 1995; Teece et al, 2002). It follows from this view of the company that ensuring organisational knowledge is effectively created, is accessible, retained and improved upon are fundamental activities for companies in order to retain competitiveness. These activities are the core building blocks of what emerged as the discipline of *Knowledge Management* (von Krogh, 1998; Quintas et al, 1997; Tiwana, 2000).

The benefits of implementing activities focusing on the knowledge of the organisation are well documented on a number of examples in the literature, as can be observed in (Nonaka, 1991), presenting initiatives taken by several Japanese companies in re-

inventing their product lines in the face of increasing competition and lowering profit returns, one example being the successful migration of Canon from its camera business to office automation products such as copying machines by leveraging expertise already present in its product development and manufacturing divisions; Another successful example can be seen at Pixar Studios (Catmull, 2008), where the fostering of a knowledge sharing culture where employees are provided forums to participate, exchange ideas and suggestions supports the creative process of new feature films. More strong evidence of the success of this approach is presented in the *Most Admired Knowledge Enterprises* award (MAKE, 2007), where global companies are chosen for their excellence in delivering knowledge processes and achieving tangible benefits in competitiveness and financial returns. In this survey it has been observed that companies present in the winning list are usually recognised leaders in their industries, and delivered twice as much return on investment than the average of Fortune 500 companies over the past decade.

## *2.2. Knowledge Management*

To support the view that knowledge is an imperative to an organisation's success, a coherent theoretical foundation and set of practices is necessary to ensure knowledge resources are effectively used, and Knowledge Management as a discipline emerged from this need. The ultimate objective of knowledge management is closely linked to the success of organisations; however the broadness of the scope and the inter disciplinary nature of the topic gave rise to a number of overlapping definitions of the term, varying on specificity, the aspects of the discipline being stressed, author preference and target audience. To illustrate this effect, and provide a better indication of the scope of knowledge management in the literature, several definitions are enlisted below:

- "*Knowledge management refers to identifying and leveraging all aspects of an organisation's knowledge to help the organisation compete*" (von Krogh, 1998).

- "*Knowledge management is a newly emerging interdisciplinary business model that has knowledge within the framework of an organisation as its focus*" (Awad et al, 2004).

- "*Knowledge management refers to processes and practices through which organisations generate value from knowledge*" (Grant, 2008).

To achieve its broad goals, knowledge management leverages the company's organisational processes, people and technology in implementing a set of tasks that map to relevant knowledge activities. To facilitate the identification of what knowledge activities are in fact needed, and what are the potential benefits obtained, a number of frameworks have been proposed in the literature. The view of knowledge activities as a workflow of coordinated stages has led to the formation frameworks based on knowledge life cycles, as seen in (Awad et al, 2004), and similarly presented in (Alavi et al, 2001). The knowledge management life cycle proposed in (Awad et al, 2004) entails four knowledge activities that can be executed iteratively throughout the organisation: first, there is *capturing* organisational knowledge from various sources; once captured, knowledge can be *organised* in more appropriate and useful ways; in the next step knowledge is *refined* and aggregated for different uses; finally knowledge is *transferred* across the organisation. Each stage comprises several sub-activities highlighting the typical concerns of knowledge management initiatives, as illustrated on Table 1. In (Alavi et al, 2001) a similar framework of high level intertwined knowledge systems is presented, and activities are divided into knowledge *creation*, *storage and retrieval*, and *transfer*.

| Stage | Activities |
|---|---|
| **Capturing** | Data entry, Scanning.<br>Brainstorming.<br>Interviewing.<br>Voice and Video input. |
| **Organising** | Cataloguing.<br>Indexing.<br>Filtering. |

| | |
|---|---|
| | Encoding. |
| **Refining** | Collaborating.<br>Contextualising.<br>Compacting.<br>Mining. |
| **Transfer** | Sharing.<br>Push / Alerting.<br>Publishing. |

**Table 1 - KM Life Cycle (Awad et al, 2004)**

Other knowledge management frameworks aim at categorising knowledge activities based on a hierarchical taxonomy, such as the one presented in (Grant, 2008) where key categories are knowledge generation and exploitation, with various sub activities indicating more granular tasks. Several other frameworks have been proposed both in research and by practitioners. In (Hoslapple et al, 2002) a proposed framework of knowledge management episodes based on previous research was evaluated and reviewed by a panel of practitioners. Frameworks have also been surveyed in the literature in (Holsapple et al, 1999) and (Tiwana, 2000), with varying approaches targeting specific problems or emphasising a specific view of the discipline.

### 2.2.1. Characterising Knowledge

Knowledge is a broad concept that has occupied the minds of philosophers and researchers for many centuries. From its roots in philosophy, the nature of knowledge has been studied in cognitive sciences, linguistics and biology (Allix, 2003). Within the somewhat narrower scope of knowledge management, the epistemological debate around the nature of knowledge is normally considered out of scope (Alawi et al, 2001; Allix, 2003; Davenport et al, 1998), and instead a more pragmatic approach is taken by investigating only perspectives that contribute to building a theory of organisational knowledge (Alawi et al, 2001; Davenport et al, 1998). Notwithstanding this, the term has received distinct definitions, highlighting the different perspectives it can take in the context of knowledge management:

- Knowledge can be seen as the accumulation of facts, procedural rules and heuristics and defined as *"understanding gained from experience and study"* (Awad et al, 2004).

- Knowledge is defined by Nonaka as *"a justified belief that increases an entity's capacity for effective action"* (Nonaka, 1994).

- In (Davenport et al, 1998) knowledge is defined as a *"mixture of framed experiences, values, contextual information and expert insight for evaluating and incorporating new experiences and information"*.

Furthermore, as noted on (Alawi et al, 2001), knowledge can be defined according to different perspectives: as a *state of mind* achieved via knowledge acquisition, as a *process* where knowledge leads to a particular action, an *object* that can be manipulated, a *condition* of having access to knowledge, or as an organisational *capability*. Focusing on a particular perspective will affect which knowledge management strategy and supporting systems will be employed. For example, an object view of knowledge may privilege strategies that emphasize the creation of knowledge stocks for the accumulation of knowledge objects.

We now discuss in more detail the perspectives on knowledge that provide a better understanding on key attributes relevant to achieving the goals of knowledge management, and to that of creating new knowledge in particular.

Tacit Knowledge and Explicit Knowledge

An important dimension to knowledge is the distinction between *Tacit* and *Explicit* knowledge (Nonaka, 1994). *Explicit* knowledge is knowledge encoded into a formal language, can be expressed numerically or in symbols, can be easily stored and transferred. This is, for example, the knowledge that exists in documentation, electronic mail, presentations and reports. *Tacit* knowledge can be seen as knowledge that exists within people's minds and is closely related to the individual's actions, experiences, commitment and involvement. Tacit knowledge comprises a technical dimension that indicates the level of know-how, as well as a cognitive dimension related to beliefs, ideals, values and mental models of an individual (Nonaka, 1998),

and because tacit knowledge is already internalised it is ready to be applied and is sometimes referred to as *actionable* knowledge (Marwick 2001).

Data, Information and Knowledge

Attempts to establish the nature of knowledge often involve the discussion of the concepts of information and data and how they relate in the framework of organisational knowledge. Investigating how these concepts relate in more detail has led to a hierarchical view with an implicit scale of values commonly accepted in the knowledge management literature (Rowley, 2007). This view also implies there are processes involved in the transformation of knowledge from a lower to a higher level in the hierarchy. In this scale the concepts of *data*, *information* and *knowledge* are commonly discussed (Davenport et al, 1998; Rowley, 2007).

*Data* can be seen as a set of discrete, unorganised facts related to an event (Davenport et al, 1998), and is widely available in organisations today in the form of recorded transactions in databases, for example, as bank withdrawal records from ATMs, or records of items sold in an on-line shop. Most companies are now capable of generating large volumes of data on all aspects of their operations; however, on its own data lacks context and relevance, and data accumulation per se will not necessarily bring positive benefits to a company (Awad et al, 2004).

*Information* can be seen as a data message intended to a receiver, aimed at improving the receiver's judgement or behaviour (Davenport et al, 1998). To the receiver, information has always meaning and belongs to a context. In this view, data is not necessarily information: a long list of bank account transaction records may be irrelevant if out of context or not used by the right person. However we can add value to data, for instance by placing it in a summarised transaction report, so that in the right context it gains meaning to a receiver and therefore becomes information.

Information adds value to data by giving it context and meaning; however information on its own will assist but does not generate better decisions or new processes and products. First, it must be put to use. Davenport defined knowledge as the "mixture of framed experiences, values, contextual information and expert insight". It is also stressed that "it originates and is applied in the minds of *knowers*" (Davenport et al,

1998), thus giving it a very human perspective. This definition also suggests information is one of many components of knowledge.

Knowledge is derived from information once information is assimilated into the mind of a knower for a relevant goal; Davenport proposes information becomes knowledge through enhancement processes requiring the *knower's* involvement, such as comparison, analysis of consequences, making connections. The progressive scale from data to knowledge has also been noted in (Zeleny, 1987), with a similar observation stating that whereas data and information may exist per se, knowledge can only exist after having being added with context, opinion and judgement by a human.

To illustrate the hierarchy arising from the analysis of the relations between data, information and knowledge, the *knowledge pyramid* diagram is commonly referred to, and displayed below. In some cases, the pyramid is added with a top layer indicating *wisdom* as the accumulation of knowledge that encompasses vision, foresight, critical thinking and the transferring of knowledge to different contexts (Rowley, 2007; Awad et al, 2004). This concept of wisdom however is not as widely discussed in the literature (Rowley, 2007), and some authors prefer a simplified approach whereby wisdom attributes are embedded in the concept of knowledge (Davenport et al, 1998). The hierarchy of the knowledge pyramid also suggests data is a more tractable entity for the purposes of encoding and programming than knowledge would be, again suggesting the higher levels of the pyramid require increasing human interactions.

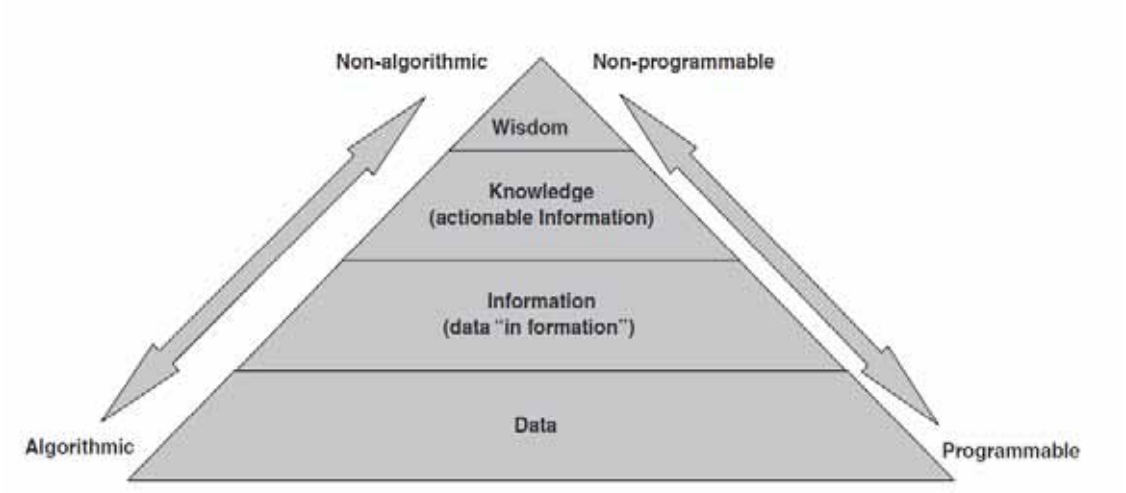



**Figure 2 - The Knowledge Pyramid (Awad et al, 2004)**

## *2.3. Knowledge Creation*

Creating new organisational knowledge is at the heart of corporate innovation. Companies must be capable of acquiring and generating new knowledge, and recycling existing knowledge into novel, more relevant ideas (Nonaka, 1995). However generating new knowledge is not a straightforward process, and it has been observed to be one of the least systematic and difficult to measure processes of a knowledge management initiative (Davenport et al, 1998). In this section, Nonaka's model for knowledge creation is presented in more details, followed by a discussion on factors affecting the creation of new knowledge within the company environment.

### 2.3.1. Nonaka's Spiral of Knowledge (Nonaka, 1994)

A theory of organisational knowledge creation was postulated by Nonaka (Nonaka, 1994) as the interplay of two dimensions of knowledge: first, there is the *tacit-explicit* dimension, where knowledge moves between its encoded explicit form into the actionable tacit form; second, there is the *ontological* dimension: knowledge can only be created by individuals and must be amplified to an organisational level in order to be effective. Permeating these two dimensions are the concepts of intention, autonomy and fluctuation as behavioural drivers which act as a medium for knowledge movement along the tacit-explicit and ontological dimensions. *Intention* can be seen as an individual's willingness to act upon knowledge, driven by personal behaviour and motivations. *Autonomy* relates to an individual or a group's degree of freedom within the organisation, and increases the possibility of unexpected opportunities to occur; finally, *fluctuation* indicates a degree of uncertainty or noise within the organisation's environment, which tend to increase chances on unusual and novel patterns to occur.

<u>The Spiral of Knowledge</u>

Nonaka introduces the idea of creating knowledge through the conversion between tacit and explicit forms (Nonaka, 1994), and postulates four modes of knowledge conversion in the context of an organisation:

- *Socialization (tacit to tacit)*: In this mode knowledge is transferred between individuals by shared experiences, without the need of an explicit representation. This is analogous to the work of an apprentice learning from his master by simple

observation and imitation. Naturally this mode requires an individual as a source of knowledge, and is normally very difficult to replicate to larger groups.

- *Internalization (explicit to tacit) and Externalization (tacit to explicit)*: In these modes, knowledge is transferred to and from an explicit form. Internalization can be compared to the idea of learning, whereas externalization transform knowledge in an encoded form that can be transmitted and replicated to other individuals with relative ease, as would be the case with training material, books and technical diagrams.

- *Combination (explicit to explicit)*: Existing resources in explicit format can be recombined to produce new knowledge through activities such as sorting, categorization and review into a different context. With the wide availability of explicit knowledge in existing computer systems, these may not always be in use to their full potential (Quintas et al, 1997) it has been noted that this reprocessing of knowledge can be of aid in reducing knowledge overload and improving knowledge based decisions (Holsapple, 2002).

The table below summarized the four modes of knowledge transformation in Nonaka's framework:

| *From / To* | *Tacit* | *Explicit* |
|---|---|---|
| *Tacit* | **Socialization** | **Externalization** |
| *Explicit* | **Internalization** | **Combination** |

**Table 2 - Modes of Knowledge Creation (Nonaka, 1994)**

If knowledge can be created upon carrying out one of the above transformations, then a dynamic process that encourages and manages such transformations at an organisational level should be aimed for. To maximize the practical benefits of such process, knowledge transfers should be encouraged continually amplified to a wider audience, thus increasing its relevance and reach, in what Nonaka had conceptualised as the *spiral of knowledge*.

### 2.3.2. The Conditions for Knowledge Creation

As seen in Nonaka's model of knowledge creation, behavioural drivers play a key role in tacit and explicit knowledge transformations, and in amplifying new knowledge throughout the organisation. The right attitude is required from individuals so that knowledge is shared instead of hoarded (intention); individuals need to act within an organisational culture that values and fosters knowledge creation and sharing (autonomy), and finally a substantial degree of uncertainty is required so that new knowledge can flourish out of experimentation (fluctuation). The individual and organisational conditions for knowledge creation are also noted in (Awad et al, 2004), where personal factors such as personality and attitude, and also vocational drivers provided by the organisation such as compensation, work environment, moral values, job security and employee recognition play a key role in knowledge sharing and creation. The requirement for a level of uncertainty in the knowledge creation process is also acknowledged in (Davenport et al, 1998), and illustrated by case studies where companies intentionally build teams with people from diverse backgrounds and personalities since different backgrounds often imply the use of different vocabularies to describe similar situations thud forcing the exchange of concepts; and different personalities which bring with them different ways of approaching a problem, knowledge exchange is thus maximised.

In order to foster knowledge creation, time and space resources must be made available so that such activities can take place. As noted in (Davenport et al, 1998), companies tend to dedicate such resources to knowledge creating activities through the creation of research centres and research and development departments. However, as illustrated by the Xerox PARC case study, care must be taken to ensure that knowledge is not only created, but is being disseminated internally in the company. In this instance, Xerox missed the opportunity to capitalise on graphical user interfaces already developed in their research centre, with Apple taking the lead in the field: an example of knowledge being created in one section, but not "spiralled" through other areas of the company.

It is also worth noting that space for knowledge creation may not always denote physical space in the form of laboratories. Knowledge sharing spaces where people

can congregate, meet and share experiences are equally as valuable (Davenport et al, 1998), as in the example illustrated in Pixar's office design that maximises opportunities for people to meet (Catmull, 2008). Space for knowledge creation may also occur electronically, facilitated by information technology tools (Marwick, 2001).

In (Nonaka et al, 1998), the need for having appropriate knowledge creation spaces is linked to the four stages of knowledge transformation in the spiral of knowledge, and formalised in a framework guided by the concept of *Ba* – a Japanese word that roughly translates to "place". *Ba* refers to a shared space or platform where knowledge creation can take place. This can take the form of physical spaces (meeting rooms and laboratories) as well as virtual spaces (typically embedded in a information technology system), or simply mental spaces (shared concepts, ideas and vocabulary). Organisations should foster the creation of *ba* in the workplace to act as facilitators of the four stages of the spiral of knowledge, and enable the transformation of information into knowledge. For each stage in the spiral of knowledge, one type of *ba* acts as the strongest influencer. At the tacit-to-tacit conversion, the *originating ba* reflects shared ideas, mental models and behavioural patterns that enable face-to-face knowledge conversions; The *interacting ba* supports tacit-to-explicit knowledge conversions, where space for dialogue where mental models can be explained and analysed is necessary; the *exercising ba* supports explicit-to-tacit conversion, it is where internalisation occurs and thus learning, simulated activities and active participation activities take place.

Of particular interest to this dissertation is the *cyber ba*, which supports the explicit to explicit knowledge transformation. Because it involves manipulation of codified knowledge, the cyber *ba* is where the role of information technology as an enabler of knowledge creation is more pronounced, providing the ability to assist the reorganisation of already stored knowledge into other forms with potentially novel uses (Nonaka et al, 1998; Alavi et al, 2001). In the next section, the link between technology and knowledge creation is examined further.

## *2.4. Information Technology and Knowledge Management*

The importance of information technology to knowledge management has been widely documented in the literature, and is seen as a crucial component for the success of knowledge management projects by (Holsapple, 2002; Alavi et al, 2001; Marwick, 2001; Awad et al, 2004). It is also pointed out in (Davenport et al, 1998) that the availability of certain technologies such as the internet and online collaboration tools such as Lotus Notes have been catalysts for the knowledge management movement. In (Alavi et al, 1999) a survey amongst companies with existing knowledge management systems across a broad range of industries revealed perceived positive benefits in employee communication, process efficiencies such as shorter problem solving times, and financial benefits related to higher profitability and shorter sales and support cycles.

Information technology comes in support of knowledge management in different ways. To map knowledge activities to their supporting knowledge management systems a framework is proposed in (Marwick, 2001) based on the knowledge conversion scheme seen in (Nonaka, 1994), illustrating the supporting role of information systems across the spectrum of knowledge activities, as shown in the below table.

| | |
|---|---|
| **Tacit to Tacit**<br>• e-Meetings.<br>• Online synchronous collaboration (chat, video conferencing). | **Tacit to Explicit**<br>• Question answering.<br>• Annotation. |
| **Explicit to Tacit**<br>• Visualisation.<br>• Video and Audio repositories. | **Explicit to Explicit**<br>• Text search.<br>• Document categorisation. |

**Table 3 - Information Systems and Knowledge Conversions (Marwick, 2001)**

The systems supporting the tacit-tacit dimension typically make shared experiences between individuals possible across geographical and temporal boundaries (Awad et al, 2004). In this dimension, information technology acts solely as a conductor for the exchange of knowledge through electronic medium, but the system has little involvement in providing or enhancing knowledge itself. Supporting systems such as

online chat, video conferencing and groupware fall into this category, and may bring considerable efficiencies on how tacit knowledge is employed inside a company. One example is documented in (Davenport et al, 1998) where video conferencing technology facilitated the remote troubleshooting of a problem in a large oil company by technicians based in different offices around the world. Another class of systems with similar concerns are *expertise locators* (McDonald et al, 1998), designed to assist in finding people with a required set of skills or similar interests inside large organisations.

In the tacit-explicit, or externalisation dimension, the transformation aims at forming a shared mental model of tacit knowledge. Collaboration systems, systems supporting brainstorming and descriptions of mental models such as mind maps, and online discussion databases fall into this category. Those systems allow the externalisation of knowledge through discussions and sharing points of view. A similar objective is attained by *expert systems* that leverage elicited tacit knowledge into explicit decision rules encoded into a system.

On the explicit-tacit knowledge transformation, systems are concerned with supporting the creation of tacit knowledge from explicit knowledge repositories, mainly by augmenting or facilitating the understanding of available data. These aims are closely related to information overload and mitigation strategies such as visualisation, summarisation and filtering techniques. Computer based learning systems are also considered by (Marwick, 2001) as enablers of explicit-tacit knowledge exchanges.

Finally, the explicit-explicit knowledge transformation is perhaps where the role of information technology is more pronounced. The increase of explicit knowledge available inside information systems also generates opportunities for supporting systems to provide knowledge combination approaches. These would include automatic document classification, summarisation and search capabilities. Other authors place the discovery of knowledge from large amounts of data within this dimension of knowledge transformation (Awad et al, 2004; Nemati et al, 2002). The discovery of new knowledge from data will be investigated in depth in the next section.

The above framework suggests a broad range of possibilities for the use of information systems to knowledge management. It is however accepted that knowledge management initiatives entail several non-technological aspects involving organisational strategy and how knowledge issues will be addressed (Hansen et al, 1999). It has been seen in the previous sections that knowledge by its very nature is ultimately personal and belongs to a human entity. Reservations are noted in the literature to approaching knowledge management initiatives strictly from the perspective of an information system implementation, where such concerns may not receive the deserved attention (von Krogh, 1998; Fahey 1998; McDermott, 1999; Prusak, 2001).

One example can be seen in (McDonald et al, 1998), where it was observed how patterns of behaviour affect the usability of expertise location systems, by illustrating how users may seek expertise not by directly reaching experts found in an expertise location system, but by using escalation procedures across the hierarchy of the organisation where political help is easier to find. Human resource factors such as employee motivation and communication have also been noted in (Hahn et al, 2000) to affect the success of a system implementation of knowledge management initiatives. Another example in (Davenport et al, 1998) illustrates how organisational changes have led to the abandonment of a knowledge management system, once executive support for the costly task of capturing expert knowledge in explicit format was no longer present.

Despite this caveat, the role of information technology as an enabler of knowledge management initiatives can be successful when applied with a clear perspective on the organisation's knowledge challenges. In (Hansen et al, 1999) several successful initiatives were investigated and a framework was devised dividing their use of information technology into *codification* and *personalisation* strategies. Codification focuses on achieving economies of scale by making knowledge explicit and widely available within the company, whereas personalization focuses its efforts on facilitating sharing of difficult to encode tacit knowledge like online collaboration systems, video and voice conferencing. Each of these strategies must be evaluated in light of the company's approach to clients, the economics of the industry, and staff profile. One example from the study presents a high-profile management consulting

firm based on small teams with flat hierarchies, very distinct projects and where customer relationships generally occur at executive level. The nature of its business determined this company should favour knowledge systems for personalisation, facilitating the tacit communication between team members for solving clients' problems, but has little to gain from mass economies of scale and reusing of explicit knowledge from repositories. Another study extended this model indicating initiatives should also take into account the *volatility* of knowledge within the industry, or how frequently knowledge changes and becomes obsolete, to assist in determining how to best employ codification or personalization strategies (Kankanhalli et al, 2003).

## *2.5. Knowledge Discovery*

With the popularisation of information technology, organisations are now capable of storing very large amounts of data in digital format, with nearly all aspects of a company's business processes being undertaken with the assistance of information systems, and recorded into data repositories, and sources of information ranging from documents, emails, company memos, customer transactions and collaboration systems exist in explicit format throughout the organisation. As seen on (Nonaka, 1994), the explicit-to-explicit, or *combination* mode of knowledge transfer is an integral step in the spiral of knowledge and contributes to knowledge creation inside the organisation. Thus, it could be argued that such vast data repositories could yield novel and useful knowledge if analysed and provided to the right (Awad et al, 2004).

However one side effect of the widespread use of information systems is the generation of very large amounts of data over time, which can be retained by long periods of time at little expense. Due to the sheer volume of data available, explicit knowledge combination work such as categorization, summarization and data analysis becomes a time consuming effort, sometimes impossible do be executed manually. Not being able to extract information from the available data repositories may lead to inefficiencies in productivity, cause poor decision making by not availing of the most accurate and most recent information, and may ultimately affect employee motivation (Cody et al, 2002; Farhoomand et al, 2002). Thus, to fully take advantage of large scale data repositories available today, automatic methods that employ the computing

power of today's information systems are required. These are challenges addressed by the area of research of *knowledge discovery in databases*, or *data mining*.

### 2.5.1. Knowledge Discovery and Data Mining

Knowledge discovery is the product of scientific research in mathematics, computer science, statistics and engineering, coupled with advances in information systems which precipitated in high volume data being easily stored and accessible. It also emerged from the need of businesses to be more competitive and accelerate the process of creating new knowledge and innovating (Awad et al, 2004).

Using knowledge discovery methods, data can be analysed using descriptive techniques to obtain more understandable or summarized representations of data, which may highlight yet unseen patterns, rules and relationships in the data (Fayyad, et al, 1996; Hand et al, 2001). Prediction is also a common task in knowledge discovery, where algorithms can learn patterns embedded on large data sets, and applied to predict future occurrences such as the anticipating fraudulent bank transactions. Data can be further analyzed using interactive exploratory methods that enable knowledge analysts to establish relationships visually that would otherwise not be possible on extremely large volumes of data (Hand et al, 2001).

Knowledge discovery started attracting significant interest in the research community during the early to mid 1990s, amid the rise in importance of knowledge discovery activities within organisations, and growing availability of collected raw data from a variety of information systems. During this time the first definitions of knowledge discovery in databases appeared in the literature. The term itself emphasises *knowledge* as the end product of the process. In (Fayyad, et al, 1996), the term is defined as follows:

> "*The nontrivial process of identifying valid, novel, potentially useful, and ultimately understandable patterns in data*".

In the knowledge management literature, similar definitions for this process occur with different names. the terms "knowledge discovery", "data mining" and "business

intelligence" are sometimes used interchangeably, with knowledge discovery being more popular in academic research, whereas data mining is a more widely used term amongst practitioners in the industry (Piatetsky-Shapiro, 2007). In (Awad et al, 2004) a definition of data mining in the business context is given as follows:

> *"Data mining is producing knowledge and discovering new patterns to describe data. DM is also predicting future values and business behaviour"*.

However, according to (Fayyad, et al, 1996), at the heart of the knowledge discovery process is the application of automated methods that can extract useful patterns from large volumes of data, referred to as *data mining algorithms*. In this context, data mining refers to the step in the overall discovery process whereby algorithms are applied to data. For consistency, the term will be used in accordance to the above definition throughout this dissertation.

### 2.5.2. Implications to Knowledge Management

The discovery of new knowledge from explicit repositories employing automatic methods is regarded by some authors as a key knowledge management activity of the modern organisation, and a crucial one to knowledge creation. In (Herschel et al, 2005), business intelligence initiatives that apply automated methods to support data analysis is seen as a crucial component in decision making and the creation of corporate knowledge; A framework that encompasses discovery activities with knowledge management is proposed in (Wang et al, 2008). The knowledge creating aspects of knowledge discovery and their relationship to knowledge management are also highlighted in (Awad et al, 2004) and (Gargano et al, 2008).

In the context of knowledge management, it has also been noted that knowledge discovery can assist not only on the knowledge creation process but also in support of other knowledge management activities such as organising knowledge (Wei et al, 2002) and performing knowledge elicitation tasks automatically (Awad et al, 2004). One example of such use is the eClassifier system for document exploration, categorization and taxonomy construction presented in (Cody et al, 2002).

## *2.6. Conclusion*

This chapter has investigated how changes in the competitive landscape had influenced the knowledge oriented view of a company, and the subsequent changes in organisation strategy. In particular, increasing competition and the need for constant innovation has increased the perception of knowledge as a strategic asset for a company's success.

Knowledge management concerns the management of all aspects of an organisation concerned with creating, retaining, renewing and applying knowledge to a company's benefit. To achieve its goals, several frameworks capturing relevant knowledge activities were proposed and surveyed. Also, perspectives in knowledge that fit the needs of knowledge management were presented. The tacit-explicit dimension and the hierarchical view of data, information and knowledge are most notably contributors to the view of knowledge within the knowledge management literature.

The knowledge creation aspects of knowledge management were further investigated in this chapter. Creating new knowledge within the scope of an organisation requires the correct conditions to be present, amongst others the ability to perform knowledge conversions along the tacit-explicit dimension, and allowing for creative conditions to exist inside the company, such as opportunities for knowledge sharing, and physical and virtual knowledge creation spaces.

Knowledge discovery provides a process and methodologies for unearthing useful information from large sets of data that would not otherwise be feasible within reasonable time frames. The motivation for performing knowledge discovery comes from the potential for creating new knowledge by extracting novel patterns from existing information already stored in explicit format and now widely available on information systems across the organisation.

Knowledge discovery is an important component to creating new knowledge in the organisation by means of transforming explicit knowledge into new explicit knowledge, as described in the *combination* knowledge conversion mode on Nonaka's spiral of knowledge (Nonaka, 1994). Applying knowledge discovery is thus beneficial

to the overall creation of knowledge inside an organisation, and is seen as a crucial component of knowledge management initiatives (Wang et al, 2008; Awad et al, 2004). It is worth highlighting that this conclusion also applies to the particular case of this dissertation's experiment, where methods for automatic discovery of opinion information from text are investigated, and can form an important component on the decision making process of certain organisations.

In the following chapter, processes for knowledge discovery are investigated in more details. The key challenges facing discovery projects are discussed, and a detailed survey of discovery activities is presented, illustrating in more details what type of new knowledge can emerge from data by applying data mining techniques. Predictive methods are investigated in more details, and a survey of the state of the art in applications for performing knowledge discovery is presented.

# 3. KNOWLEGDE DISCOVERY AND DATA MINING

This chapter explores related research on the methodologies, goals, key challenges and available tools for the execution of a knowledge discovery project. Key knowledge discovery processes developed by research and the industry are discussed, and what are the important success factors of a project. Data mining techniques presented in the literature are reviewed, highlighting the objectives, potentials and challenges of each technique. The chapter continues with a more detailed review of classification in data mining, algorithms used in classification, and discusses important aspects of classification relevant to the opinion mining experiment performed as part of this dissertation. To conclude the chapter, a discussion on applications for executing data mining projects is presented.

## *3.1. Knowledge Discovery Processes*

Knowledge discovery is a complex activity involving multiple steps and requiring diverse abilities, such as skills coming from individuals with business understanding, analysts possessing familiarity with the data, information technology professionals and data miners. As with any complex undertaking, a systematic approach to performing all required tasks is crucial to ensure projects are successful, and that successful projects are repeatable. This need has not gone unnoticed on both research circles and in the industry, with six different methods having been identified and surveyed in (Hofmann, 2003). The KDD process and its reviewed version (Fayyad et al, 1996; Collier et al, 1998) are an important methodology for knowledge discovery coming from academic research. In the industry, the need for a common framework for performing data mining that embedded best practices from companies and practitioners resulted in the CRISP-DM methodology (Chapman et al, 2000); in addition, core knowledge discovery processes are also embedded in software offerings by industry providers, as is the case with SAS SEMMA data mining technique of *Sample, Explore, Modify, Model and Access* (SEMMA, 2008). In this section the KDD and CRISP-DM knowledge discovery processes are discussed in more details, the approaches are compared with further considerations on the important aspects of a knowledge discovery project. These considerations will be of importance to this dissertation's

experiment, which will perform a data mining task, and will benefit from best practices embedded on these methodologies.

### 3.1.1. Analysis of Knowledge Discovery Processes

To further understand the activities involved in performing a knowledge discovery project, this section discusses two processes originating from academic research and the industry: the KDD process for knowledge discovery in databases proposed in (Fayyad et al, 1996), and CRISP-DM processes (Chapman et al, 2000) originated from a consortium of companies involved in data mining.

**KDD Process (Fayyad et al, 1996)**

The KDD Process is a series of interactive steps to achieve the goal of finding useful knowledge from large amounts of raw data. The process is designed to be iterative: any sequence of steps may be refined and re-executed several times. The diagram below illustrates the main stages of the KDD process.

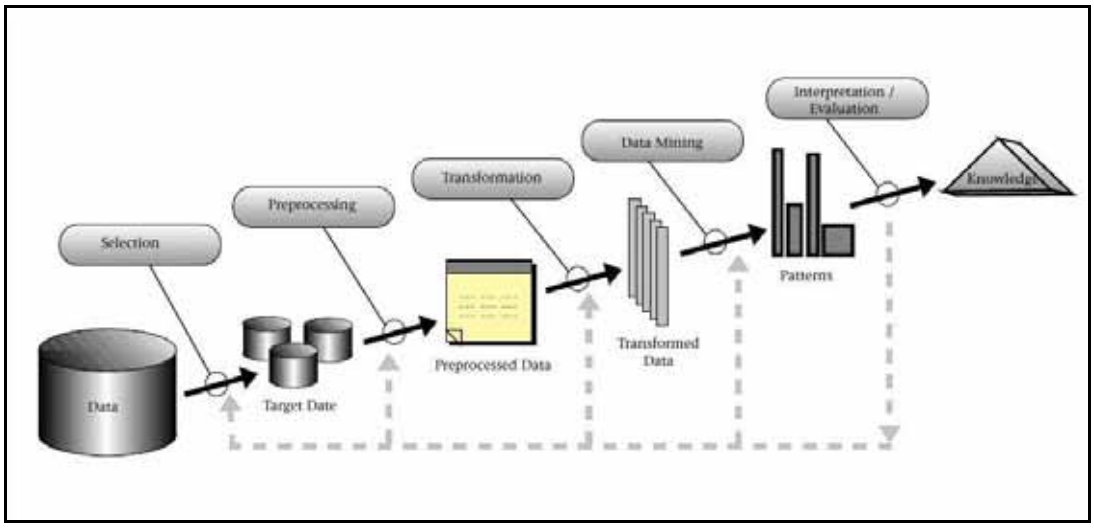



**Figure 3 - Stages of the KDD Process (Fayyad et al, 1996)**

The process begins with *identifying the problem domain and business goals*, where business users and data analysts discuss requirements, scope and how to approach the problem from a data mining perspective. Next, work begins on the *creation of target data set* to be analyzed, and data from all relevant sources is identified and sourced.

Once a data set is produced, *data cleansing and pre-processing* occurs: here, important pre-requisite steps are performed such as removing noisy data, handling missing values and outliers and data type correction, which may impact the performance and quality of the final result. The next step in the process is *data reduction*, where a subset of the overall data is selected based on its relevance to the data mining task. Data reduction is of critical importance in speeding up mining algorithms to acceptable performance levels, especially in cases where data may contain a large number of attributes, or the data set is in the order of several million records.

After data cleansing and reduction, a data set is produced and ready to be mined, and work can begin on *choosing a data mining task*, based on discussions with business users and project goals. Also, *exploratory analysis* of the data set can take place; this provides further insights on the nature of the data, helps in determining the data mining tasks and provides early feedback on collected data to business users. Any corrective action can take place by re-executing earlier stages on the process. Finally, the *data mining* stage is executed, where data mining algorithms are applied to the data set based on criteria determined on previous stages. Then, *results are evaluated and interpreted*, and it is likely that this will lead to several iterations of the mining stage, so that algorithms can be fine tuned and hypothesis can be confirmed. Lastly, by reviewing results of the data mining exercise with business users, these can be used as new knowledge and *acted upon* in their relevant business context.

**Revision of the Original KDD Process (Collier et al, 1998)**

It was observed in (Collier et al, 1998) that the original KDD process touched only briefly on two important aspects of knowledge discovery: the *framing of data mining questions*, arising from business requirements but specifically targeted at directing the data mining modelling work, should be part of the initial stages of knowledge discovery. The second aspect is *actionable results*: it is not enough to provide discovered patterns as the output of a discovery exercise. Instead, directions on how the discovered knowledge will be put to practice in the business context should be included for a more transparent assessment of its benefits. The iterative nature of the process is also a strong aspect of knowledge discovery, and should be made more evident. A new diagram is proposed illustrating these remarks:

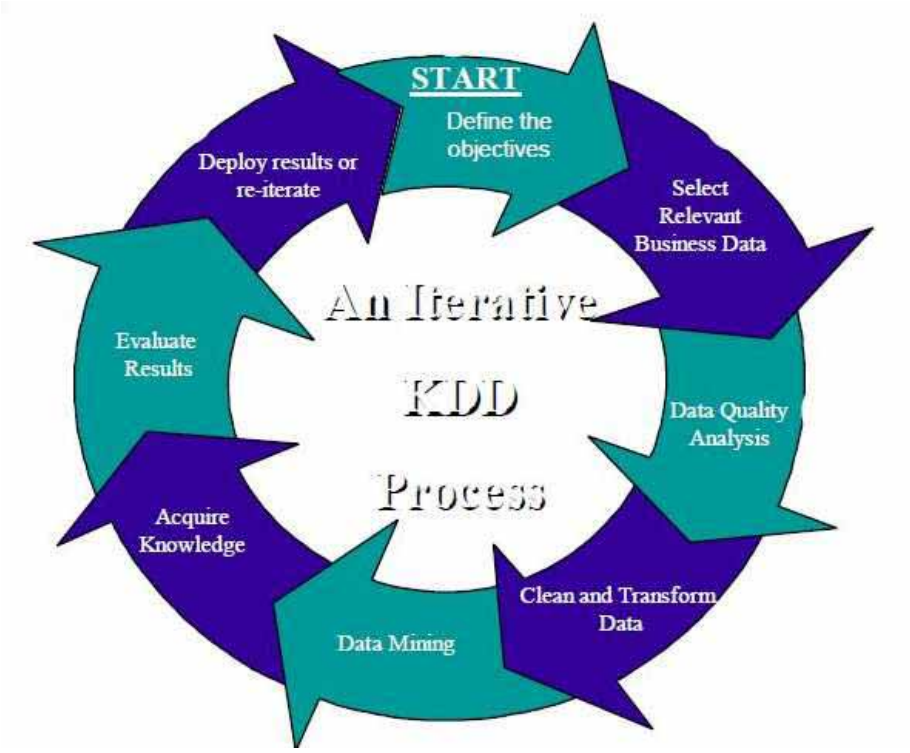



**Figure 4 - Revised KDD Process by (Collier et al, 1998)**

**CRISP-DM (Chapman et al, 2000)**

The CRISP-DM methodology (Cross-Industry Standard Process for Data Mining) grew out of the need to have a common platform-independent process model for implementing data mining projects on an industry that was experiencing an explosion in demand. It started in 1996 as a consortium formed initially by pioneer companies heavily involved in the field of data mining – Daimer-Benz, NCR and SPSS – that later transformed into a special interest group comprising several hundred representatives from the industry with a stake in data mining technology. Two and half years later the initial draft version of the CRISP-DM 1.0 methodology was published (CRISP-DM, 2000b).

CRISP-DM is a process model describing data mining activities in four hierarchical levels of abstraction. At the top level are the project *stages*, which generally describe a data mining project and can guide implementations, but can easily be transported between industries and data mining scenarios. Any stage can be drilled down into

*generic tasks*, *detailed tasks* and *process instances*, which describe with in high detail what each activity entails, what are their input and expected outputs. In this regard CRISP-DM is a more comprehensive methodology than the high level steps outlined in KDD. CRISP-DM is both a reference model and a user guide, embedding practitioner's knowledge and best practices learned from past projects. The CRISP-DM project life-cycle is divided into iterative stages, but with no specific execution sequence. Each stage can be repeated depending on obtained outcomes from previous stages, nature of the data and data mining objectives. These are outlined below along with a diagram illustrating key interactions between stages:

- *Business Understanding*: the initial stage where business requirements are understood and agreed upon, a definition of the problem is devised between data miners and business analysts and planning can take place. This step will also highlight whether the data mining approach is the best or the only viable alternative for addressing the underlying business problem (Shearer, 2000).

- *Data Understanding*: this stage comprises tasks to obtain an initial data collection and familiarization with the data. An exploratory analysis of data is also part of this stage, which will generate initial findings and insights to be further developed on future stages of the project.

- *Data Preparation:* With a better understanding of the problem being tackled and nature of the data available, data can then be selected, cleaned and pre-processed in a variety of ways so that a final data set ready to be applied to a data mining technique can be created.

- *Modelling*: In this stage a data mining technique will be chosen, applied to the final data set and the results assessed. Choosing a data mining technique will depend on the nature of the data and specific requirements of the project.

- *Evaluation*: The evaluation stage provides a checkpoint to ensure the work produced so far is indeed of relevance to the project's business objectives. A more

careful and comprehensive evaluation of the results is carried out, and next steps in the exercise are agreed upon.

- *Deployment*: Finally, with the correct data and models ready, it is possible to deploy the data mining project to the wider business audience and monitor its benefits. The entire project is also reviewed with significant insights, pitfalls and lessons learned discussed for use in future projects.

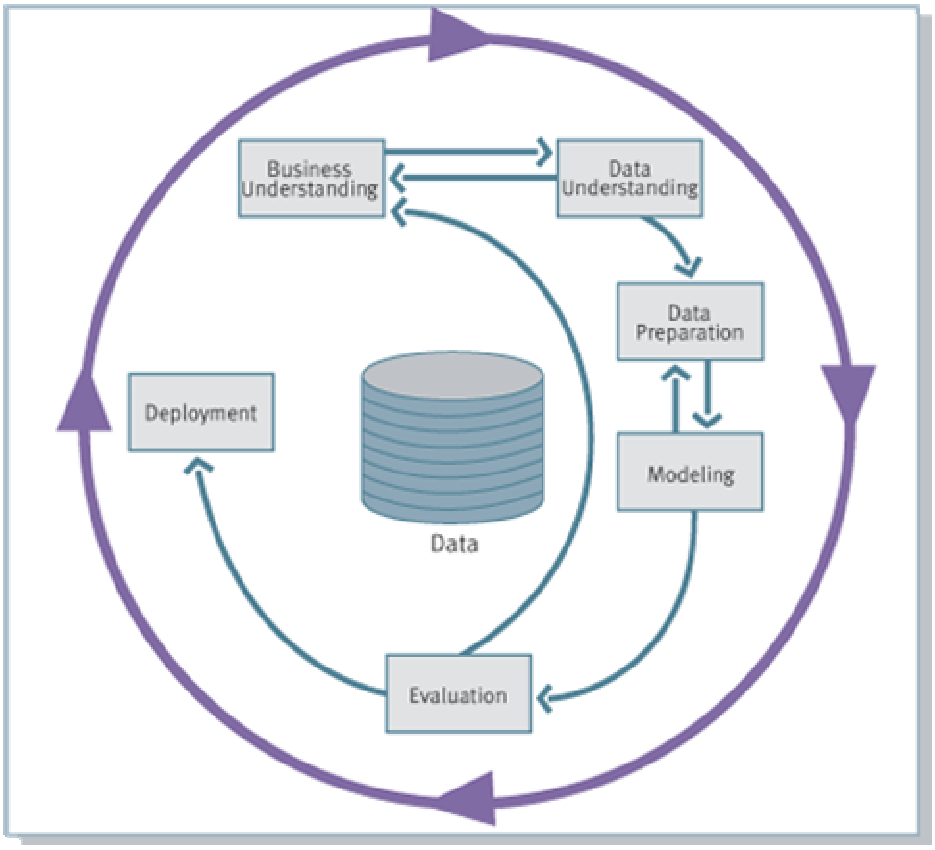



**Figure 5 - Stages in the CRISP-DM Methodology (Chapman et al, 2000)**

### 3.1.2. Considerations on Knowledge Discovery Processes

From observing the models analysed in the previous section, certain common characteristics and key concerns of performing a knowledge discovery exercise become evident: Firstly, we identify the *iterative nature* of the process being stressed on both KDD and CRISP-DM methods, suggesting that great level of flexibility is required on such projects, since the scope of the exercise is likely to be modified,

expanded or contracted depending on what is initially discovered from the available data. This can also be seen as a challenge, since it is not known a priori the number of iterations a project will require, or when the outcomes will provide a good enough answer to the originating business needs. Such decisions need to be weighted against the project's cost and timeframe constraints.

Another important point highlighted on both processes is the existence of two groups with stakes in a knowledge discovery process and distinct skill sets: *data miners* who perform the data extraction, modelling and execution activities, and business community users who understand the business problem and wish to apply the results of the process to their advantage. To be effective, the project must ensure a constant *dialogue between data miners and business users,* and loosing sight of this interaction could cause a misalignment on what the business expects and what patterns data miners unearth from data. This could become a key factor in failures on knowledge discovery projects (Pyle, 2004). Whereas both KDD and CRISP-DM acknowledge and provide stages for discussions with business users within their processes, some authors argue that this point deserves greater attention at organisational level, and that it requires the development of competencies that blends analytical skills with business acumen (Kolyshkina et al, 2007). This issue was also noticed in (Wang et al, 2008), and it is framed in a knowledge management context where the traditional knowledge discovery process cycle - typically executed by data miners - is enhanced with a second *knowledge development cycle* executed by the business community. This new cycle starts with knowledge sharing of results from the data mining exercise, and comprises learning from its results, acting and internalising the acquired knowledge and providing feedback for future mining tasks. The next diagram illustrates the two cycles and how they interact.

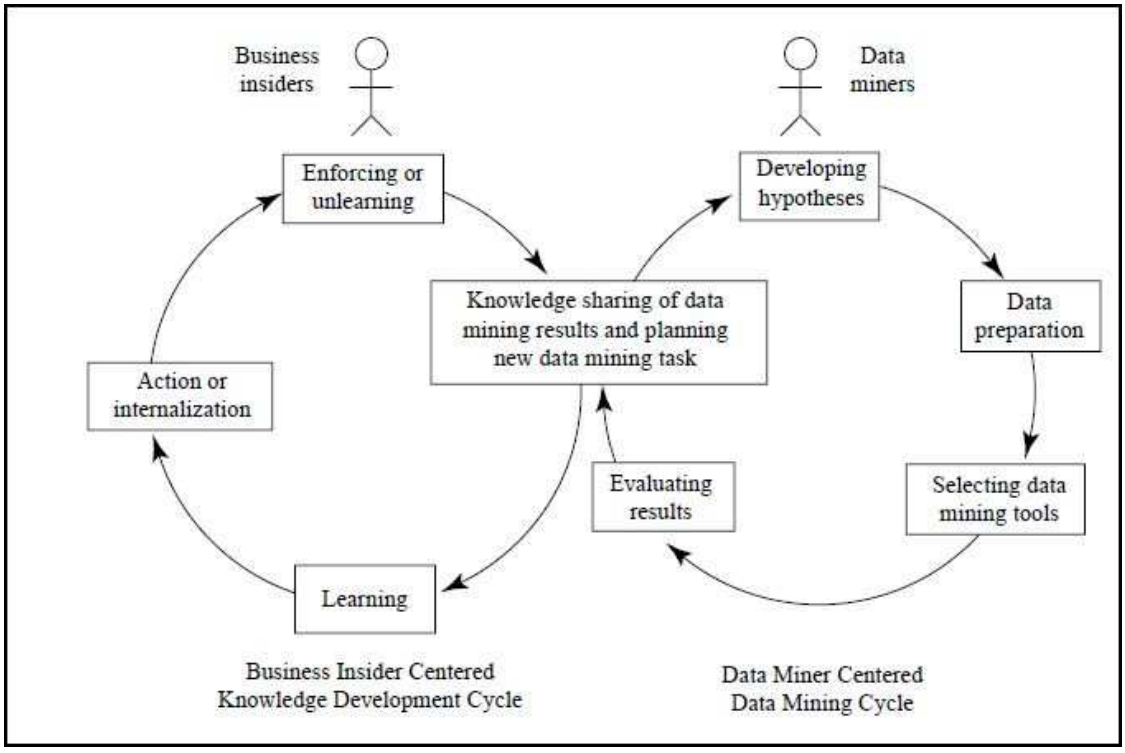



**Figure 6 - Cycles of Knowledge Development through Data Mining (Wang et al, 2008)**

The diagram also illustrates how knowledge discovery is in fact a knowledge creation activity in the context of knowledge management investigated in Chapter 2: the explicit knowledge acquired from raw data via data mining can be seen as a knowledge combination transformation in terms of Nonaka's spiral of knowledge (Nonaka, 1994), thus enabling further knowledge creation activities to occur. These are present on the knowledge development cycle pertinent to business users: new knowledge is shared, learned, acted upon and finally unlearned or enforced depending on the outcome obtained from the new knowledge. Once new knowledge has gone through the entire cycle it can be used as feedback for a new data mining cycle.

One important stage of a knowledge discovery task present on both processes, but often overlooked is *data preparation*. Collecting, cleaning and reformatting data so it can be fit for use in a data mining algorithm is a time consuming and complex task, and it is estimated that between 50% and 70% of the time in a knowledge discovery project can be spent on these activities (Shearer, 2000). For this very reason it can be tempting to reduce project costs and scope by way of neglecting or oversimplifying data preparation issues, with negative consequences to the quality of the final results (Pyle, 2004). An example in (Kolyshkina et al, 2007) illustrates how poor data

preparation can damage the outcome of a project: it was discovered after reviewing poor results from the data mining exercise, that in incorrect assumption on the source data caused a salary field to be filled with zero value on a significant number of cases, thus incorrectly skewing the final results and invalidating the analysis.

Finally, *choosing a data mining technique* for a specific problem is also a difficult task that needs thorough understanding of the project objectives and the universe of tools available (Wang et al, 2008). Choosing a model best suited for a project's desired outcome has to be done taking factors such as data availability and quality, how much understanding from the model is required for business users, time constraints, familiarity with a technique and the business goals of the project. In (Chapman et al, 2000) the CRISP-DM methodology states that initial results from a model should be assessed and tuned iteratively. CRISP-DM also provides a categorization of techniques in term of mining tasks, in order to assist in choosing the right technique for a specific problem, and in (Pyle, 2004) it is suggested that data mining goals should be extracted from business goals via the construction of a goal hierarchy, starting with the business problem being addressed and adding more layers of detail until a business question can be framed in data mining terms.

The experiment performed as part of this dissertation comprises the execution of a data mining algorithm to identify opinion bias in text documents. It is useful to approach this task with a systematic methodology that will guide the execution of the experiment, embedding previously acquired best practice knowledge on common pitfalls and areas of concern, and this can be achieved by leveraging the methodologies discussed in this chapter, since the experiment will face similar issues on data preparation and choice of data mining techniques, will be iterative in nature, and should never loose sight of the higher level goals an opinion mining project hopes to achieve. In the next section the stage where data mining algorithms are applied to data is discussed in more details, with particular consideration to predictive data mining techniques for data classification – the technique used in this dissertation's experiment.

## *3.2. Data Mining Techniques*

The overall goal of data mining is to extract knowledge from large volumes of raw data that satisfies the criteria of novelty, usefulness and intelligibility (Fayyad, et al, 1996), and as discussed in the previous section, a systematic process can be applied to find, extract and prepare data so it is ready to be explored.

Data mining is a field rich in techniques available to the analyst, each with a varying number of parameters to choose from, and deciding on the most suitable one for a given task can be particularly daunting. It is also noted in (Chapman et al, 2000) that the data miner needs to choose from the universe of available tools, the ones that fit both business constraints and political requirements of the project, such as the delivery time and intelligibility of results. Deciding on what data mining techniques to apply to the data based on the overall goals of a knowledge discovery project is the goal of this section.

### 3.2.1. Goal Based Categorisation of Data Mining Techniques

To help in better choosing which data mining technique to apply, it is useful and common in the literature to categorise them according to the overall goals of the knowledge discovery projects. In (Fayyad et al, 1996), data mining methods are grouped into *prediction* and *description.* In (Hand et al, 2001), *exploratory data analysis, information retrieval* and *rule discovery* are also added to the categorisation. A similar categorisation of data mining techniques according to types of problem is also present in the CRISP-DM methodology (Chapman et al, 2000). We explore in more details the data mining goals and their associated techniques in the remaining of this section.

**Predictive Methods**

Predictive methods attempt to determine future values of a variable of interest by learning from data available on a given data set. This variable can be, for instance, the suitability of a loan application given an applicant's financial data, or the future share price for a given company. When using predictive methods, there are two important assumptions to consider: It is assumed that a data set containing instances with values for the variable we are trying to predict is available for training. Learning in predictive

methods works solely by observing past occurrences and attempting to find patterns on it, and have no ability to reason or derive conclusions from basic principles (Weiss et al, 2005), thus it is also assumed that inherent to the data set are patterns which can fit future occurrences of the information being predicted (Alpoydin, 2004). If future occurrences are completely different to the ones used in training, or the data does not capture important aspects that determine future behaviour, then prediction results may not be relevant. With this in mind data mining algorithms can be applied to "learn" such patterns and make predictions on a given variable to yet unseen occurrences, not present on the original data set. Algorithms performing predictive methods that take into account known instances of the target variable for training are categorised in the machine learning literature as *supervised learning* methods (Nilsson, 1996; Alpoydin, 2004).

Another factor to be considered is the type of variable being predicted: the target variable could be a real valued number of interest to the problem, such as predicting the target price of items in an auction; or the target variable can be determining whether an instance belongs to a pre-determined class a priori, with no numeric relevance, such as predicting whether a transaction is fraudulent, or whether an email is considered spam. *Classification* methods attempt to learn a function that maps a data into one of a set of predefined classes. There are numerous examples of the application of classification methods in knowledge discovery literature, such as the case studies surveyed in (Wei et al, 2002) for predicting credit application adequacy, customer profiling, and disease screening; and the classification of astronomical objects seen in (Fayyad et al, 1993). Other popular classification examples involve spam detection (Drucker et al, 1997), and fraud detection in financial services and telecommunications industry (Awad et al, 2004). To illustrate the goal of classification, the graphic below represents a two-attribute data set of bank loan applications labelled "yes" and "no". The grey line separates the examples into a classification boundary, and represents how future instances might be categorised:

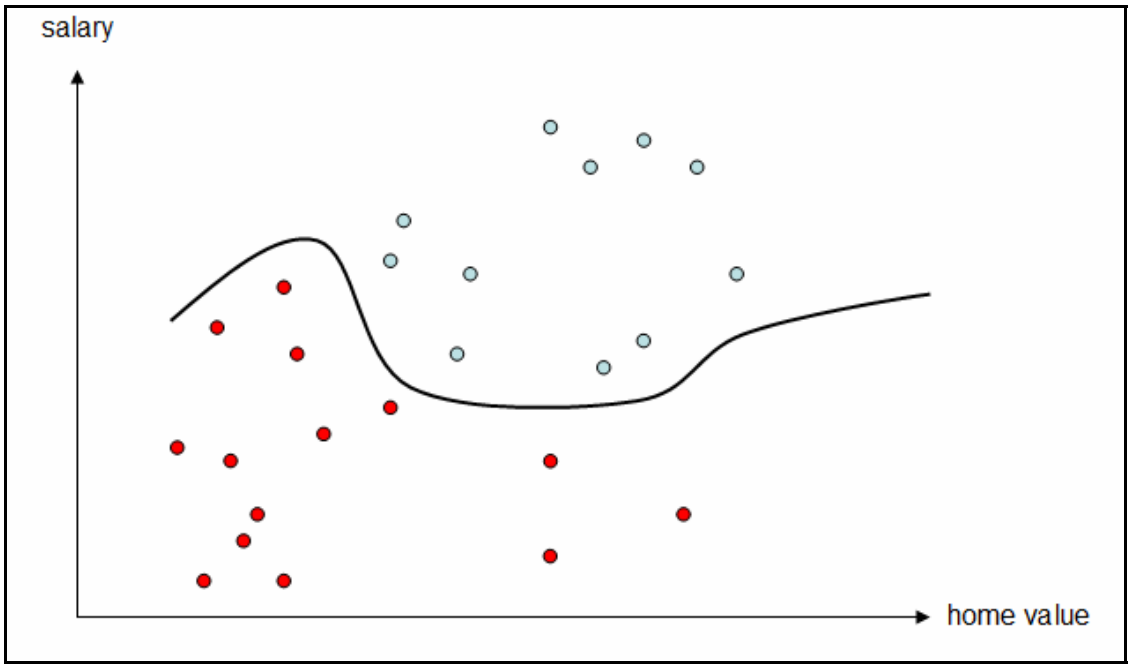



**Figure 7 - Example Classification of Loan Applications**

In *Regression* a function is learned from an initial set that maps a data item to a real valued prediction variable. The key difference between regression and classification is that the predicted value does have numerical significance (Hand et al, 2001), such as predicting future property values, fuel consumption of an automobile, or stock prices; whereas in classification the predicted class is solely a class identifier determined a priori with no numeric value.

**Descriptive Methods**

The primary objective of descriptive methods is to describe important aspects of the data set in a human understandable format. Examples of descriptive techniques include *clustering*, where data is partitioned into groups according to a similarity criteria, modelling the *probability distribution* of the data; *dependency modelling*, where models attempt to find dependency relationships between data items, and *change detection*, where models are built to find out most significant changes on data from previous measurements. Additionally, *summarization* of data in a more concise format is also viewed as a descriptive method in (Fayyad et al, 1996).

Descriptive clustering techniques have been employed to better understanding of customer transactions in the web (Yang et al, 2005) and for detecting user

personalisation preferences (Mobasher et al, 2002). Hierarchical clustering applied to categorising concepts in text data sets is seen in (Fry et al, 2008).

**Exploratory Data Analysis**

Exploratory data analysis is the application of data mining techniques to obtain further insights on the data, without a specific goal in sight. It is seen in (Fayyad et al, 1996; Chapman et al, 2000) as a preliminary stage in the knowledge discovery process aiming at acquiring familiarity with the data and directing further discovery activity, and was noted to be a crucial, often overlooked step in the success of knowledge discovery projects (Pyle, 2004; Kolushkina et al, 2007). Data exploration in itself can also be seen as a genuine data mining goal since its output may produce sufficient information to enable the discovery of knowledge and improved decision making (Hand et al, 2001).

Exploratory data analysis techniques tend to be interactive and visual in nature, highlighting the strong level of human involvement in such activities. These techniques are also named *visual analytics* in the literature, and suggest a more prominent role for human cognitive and perceptual processes in discovering patterns by interacting with data not only by using well known plotting and summarization methods such as charts and scatter plots, but also by means of innovative visualization techniques (Keim et al, 2007). To this end, more sophisticated interactive computer-aided interfaces can be employed to enrich the data exploration process by supporting typical tasks of visual analytics such as presenting a data overview, zooming, filtering and presenting details on demand (Keim et al 2007, Schneiderman 1996; Fry, 2008). Visual analytics applied to large data sets is an active area of research, with results being applied to the analysis of social networks (Kang et al, 2007), genetic pattern identification (Hochheiser et al, 2003) and text analysis (Zheleva et al, 2007).

**Rule and Pattern Discovery**

This class of algorithms is concerned with detecting patterns in data, such as regions or events that differ significantly on the data set being analyzed. Examples of applications of pattern detection include detecting fraudulent behaviour in credit card transaction, or detecting astronomical objects with unusual characteristics. Rule learning methods not only detect patterns, but often have the ability to present intelligible results that can be easily analysed (Hand et al, 2001).

One popular technique for finding rules in data sets originates from the problem of market basket analysis, where the objective is to identify from a data set of shopping transactions what items are commonly purchased together. This problem can be generalised to other areas where the objective is to identify common associations between items in a data set, and became known as *association rule mining* (Hipp et al, 2000). A large quantity of algorithms was developed to address this type of problem, the *APriori* algorithm being one of the most popular (Agrawal et al, 1994), with several improved variations being developed since its inception, along with different techniques being developed, as surveyed in (Hipp et al, 2000). Association rule mining has been put to practice on databases from several different industries, such as retail organisations looking for cross-marketing and product placement opportunities (Brijs et al, 2000), profiling students in educational institutions (Ma et al, 2000), and predicting the occurrence of heart disease given certain diagnostic conditions (Ordonez, 2006).

*Rule induction* methods describe another set of techniques aimed at representing discovered patterns in data according to a rule framework. In this context, a rule can be described in terms of a first order logic proposition, as per the example:

**If** Salary is higher than €30000 **AND** HomeOwner **then** Approve Loan.

The core idea of rule induction is to perform a search for potential rules on the existing data, and rank these rules according to a fitness function, such as rule probability or degree of generalisation (Hand et al, 2001). To this end, decision tree methods such as the algorithms described in (Quinlan, 1986) can be employed to find useful rules: a

branch is considered a candidate rule that can then be scored according to the desired fitness method. Other methods employed include search the result space with heuristics, as used by the CN2 (Clark et al, 1989) and the HYDRA algorithms (Ali et al, 1993). Examples of applying rule induction methods have been documented in the literature, for instance in systems assisting in genetic-based diagnostics (Livingston et al, 2003), and in building of stock portfolios for financial engineering (John et al, 1996).

### 3.2.2. Summary and Considerations

The table below summarizes the data mining methods according to their goals:

| **Goals** | **Data Mining Techniques** | **Application Examples** |
|---|---|---|
| Prediction (Fayyad et al, 1996; Hand et al, 2001) | Classification. Regression. | Credit application, customer profiling and disease screening (Wei et al, 2002); categorization of astronomical observations (Fayyad et al, 1993); spam detection (Drucker et al, 1997); fraud detection (Awad et al, 2004). |
| Description (Fayyad et al, 1996; Hand et al, 2001) | Clustering. Summarization. Dependency Modelling. Change Detection. Probability Estimation. | Summarisation of documents (Weiss et al, 2004); Analysis of web transactions (Yang et al, 2005); Hierarchical clustering (Fry et al, 2008); Web personalisation (Mobasher et al, 2002). |
| Exploratory Data Analysis (Hand et al, 2001) | Visual Analytics. Summarization. | Social network analysis (Kang et al, 2007); Genetic pattern identification (Hochheiser et al, 2003); Text analysis (Zheleva et al, 2007). |
| Rule and Pattern | Association rule mining. | Retail (Brijs et al, 2000); |

| | | |
|---|---|---|
| Discovery<br>(Hand et al, 2001) | | Student Profiling (Ma et al, 2000); Heart disease diagnostic (Ordonez, 2006). |
| | Rule induction. | Generating investment portfolio (John et al, 1996), and Genetic-based diagnosis (Livingston et al, 2003). |

**Table 4 - Data Mining Techniques Categorised by Goal**

The goal oriented classification of data mining techniques gives a good overview of what data mining is capable of achieving which, coupled with understanding of business objectives and domain knowledge is a useful guide in determining what methods to apply to a specific task. It can also be noted that the data mining techniques need not necessarily be used independently, and that some techniques are well suited to more than one goal. We see for instance, how similar pattern discovery methods can aid in a prediction objective (John et al, 1996) and also in data description (Brijs et al, 2000). The goals of data mining also may overlap, as rule discovery is closely related to the goal of prediction, and indeed certain predictive methods do provide explanatory capabilities suitable for a rule discovery exercise, as will be further detailed on the survey of classification algorithms in the following section.

The goal of this dissertation experiment is the prediction of sentiment orientation on text as being positive or negative, which involves the use of supervised learning methods. In the following sections two key elements for performing a classification task in data mining are defined and discussed in more details: these are the aspects of the data set and classification algorithms.

### 3.2.3. Considerations on the Data Set

One important aspect data mining is that it relies on available data for the application of techniques and extraction of meaningful conclusions. Data sets from the real world however will rarely be in the correct format for data mining, or free from a variety of errors and noise. Ensuring data represents as closely as possible the domain in which knowledge needs to be extracted, and that the quality of data has been verified is

therefore crucial to obtaining better results. Before discussing data mining algorithms in more details, it is useful to qualify more formally what is meant by the term *data set*, and discuss data quality issues commonly found in them.

At a general level, a data set comprises "*a set of measurements taken from some environment or process*" (Hand et al, 2002). The term is formalised as a set of *n* objects, for which *p* measurements on different events or attributes were made, forming thus a *n x p matrix* of data points. Each group of measurements for a given object in this matrix is commonly called a *case*, *entity*, *record*, or *vector* while the individual measurements for a given object are typically called *variables*, *attributes*, *features* or *fields* (Hand et al, 2002; Nilsson, 1996). The table below is an example of a simple data set for bank loan applications in the form of a matrix of 5 measurements and 9 attributes:

| **ID** | **Age** | **Employment Status** | **Years Employed** | **Salary** | **Dependents** | **Home Owner** | **Home Value** | **Approve Loan** |
|---|---|---|---|---|---|---|---|---|
| 100 | 44 | Employed | 15 | 35000 | 2 | Y | 250000 | Y |
| 101 | 32 | Self-Employed | 8 | 28000 | 2 | N | | N |
| 102 | 27 | Employed | 5 | 25000 | 0 | N | | Y |
| 103 | 53 | Unemployed | 21 | 44000 | 3 | Y | 300000 | N |
| 104 | 33 | Self-Employed | 9 | 35000 | 1 | Y | 20000 | Y |

**Table 5 - Example data set for loan applications**

From the above example a few key observations can be made on the nature of data sets. Firstly, for the purposes of data analysis, data can be distinguished into *numerical* – where attributes possess real values; and *categorical*, where attributes can only take a predetermined set of discrete non numeric values. In this example, "employment status" and "home owner" are categorical, as they only allow values from a specific set. The presence of numerical or categorical data will determine what algorithms are more suited for the data mining task.

Another aspect of real world data sets is that, due to measurement problems or the nature of data, attributes values may be missing or unknown, as is the case on the field "home value" in the above example. The existence of missing values can deteriorate the quality of the data mining results, and several approaches have been developed to

handle the issue, ranging from ad-hoc methods aided by domain knowledge to more sophisticated approaches. At first instance, *removing data with missing values* can be attempted: in this approach, rows where missing values are found are simply removed from the data set before performing any data mining activity. However removing a large number of entries due to missing values may skew the data set and affect the end result of the data mining task.

When removing missing values is not possible, *data imputation* methods can be applied. With knowledge of the data set and domain being modelled, missing values can be "filled in" with an appropriate value, such as a fixed value: in the example data set from Table 4, "home value" may indicate the amount of collateral being offered for the loan application, and could be replaced with a constant value where it is known that the applicant is not a home owner; other similar rules can be composed with sufficient domain knowledge and knowledge of the data collection process. Another approach would be to replenish missing values with the mean value for that attribute over all instances in the data set, or over instances whose other attributes contain similar values, in case the attribute is numeric. The use of such methods however should be cautious: it is advocated in (Weiss et al, 1998) that it is preferable to perform classification without any missing values, or rather than attempting to perform the above pre-processing tasks, apply a classification algorithm capable of handling missing values simply as another value for data. Other more sophisticated statistical techniques, such as expectation minimization and predictive mean matching have also been attempted and demonstrated superior empirical results in classification tasks on instances where missing data occurs (Su et al, 2008).

Data mining relies solely on the data set being analysed, thus other aspects to data quality such as data precision and accuracy need also be taken into account when preparing a data set. Data *precision* relates to the amount of variation observed on the measurements being collected. Measurement variability can be caused by environmental factors or instrument quality. Data *accuracy* reflects how close the measurements are from their expected "true" value. Inaccurate measures can be caused by faulty measurement devices, or bias caused by external factors. In (Hand et al, 2001), the notions of *reliability* and *validity* are also discussed: these relate closely to the concepts of precision and accuracy, but the terms are applicable to social and

behavioural sciences and the issues encountered on data collection in this area. A reliable measurement means results for a given question are repeatable for the same circumstances and persons; validity relates to how relevant is the data, or questions being asked to the phenomena being measured.

**Outliers**

Outliers are abnormal instances on data due to a considerable deviation from other instances in the data set. They can be caused by errors in measurement, and may cause distortion in the model being built by a data mining algorithm, when they can be considered unrepresentative of the domain being measured. On the other hand, outliers can be sometimes sought elements in data, as would be the case on fraud detection and spam detection systems, where it is assumed the majority of occurrences will follow a given pattern, and unusual patterns are triggered by fraudulent behaviour, or spam activity. Identifying outliers is typically carried out by a comparison with the remainder of the data set, via a similarity measurement. A method involving clustering techniques is seen in (Ramaswamy et al, 2000); with other methods surveyed in (Hand et al, 2001; Knorr et al, 1998).

## *3.3. Data Mining Algorithms for Classification*

The main objective of this section is to introduce and discuss predictive algorithms that perform the data mining task of classification, also known as *classifiers*. This chapter reviews the relevant research topics in the literature that relate to the choice and evaluation of classification algorithms. A formalization of a classification algorithm is presented, alongside with historical background on various classification methods. Algorithms for classification are briefly surveyed, with emphasis on techniques more commonly applied in the literature and with greater relevance to text classification. Finally, success criteria metrics and common challenges to classification are discussed.

### 3.3.1. Introduction

The problem of data classification can be seen as an instance of the more generic problem of fitting a model to observed data (Alpoydin, 2004), a concern that exists not only on computer science, but on different fields of research. Hence, the algorithms used as classifiers have their origins on diverse fields from induction on statistics,

pattern recognition and signal processing in engineering, machine learning, information theory and artificial intelligence to the more recent biology inspired methods such as neural computing and evolutionary methods (Alpoydin, 2004; Kulkarni et al, 1998). Supervised learning algorithms are an active field of research with new techniques and improvements to already existing algorithms being developed constantly. A comprehensive survey of all available classification techniques in the literature would be beyond the scope of this dissertation. Indeed, being such a vast topic, the difficulties in compiling an exhaustive survey have been noticed as early as in (Ho et al, 1968). Nonetheless, it is relevant to present and discuss the development of well known classes of algorithms and state-of-the-art methods, and their relative strengths and weaknesses. This assessment will allow for a more careful selection of what classifier will be employed to this dissertation's experiment and will allow us to further illustrate the implications of choosing a particular algorithm over another. Well known classes of algorithms are commonly discussed in recent data mining textbooks, such as (Hand et al, 2001; Alpoydin, 2004; Weiss et al 1998) and also in more specific data mining literature such as text mining (Weiss et al, 2005). A survey on algorithms focusing on two class pattern classification including state-of-the-art methods can be found on (Kulkarni et al, 1998). Variations on specific classes of classification algorithms such as neural networks and tree-based methods are surveyed in (Zhang, 2000) and (Lim et al, 2000).

### 3.3.2. Supervised Learning Algorithms

As discussed in section 3.2.1, a supervised learning algorithm attempts to predict future values of a given variable based on information contained on an already present data set used for training. The data set contains instances of the variable we wish to predict, and it is assumed that future values retain a certain similarity to already observed values, which can be "learned" by a supervised learning algorithm. This dependency on the available data as being representative for predictions is worth stressing: if future values do not retain any similarity to already seen data, prediction results will not be reliable (Alpoydin, 2004; Weiss et al 2005). Thus, the design of good supervised learning algorithms has a dependency on the data available for training.

Judging the fitness of a classifier depends in part on the data mining objectives, and on how technical factors such as training time are likely to affect the end result of the task. In (Weiss et al, 1998) the evaluation of a classifier is based on its data pre-processing requirements, solution complexity, timing and explanatory capabilities. Data *pre-processing requirements* relate to issues such as handling of missing features and ability to handle both numerical and categorical features; *solution complexity* indicates the level of parameterisation that the underlying algorithm model allows; *timing* indicates training time and algorithm runtime, which when considering large volumes of data are important factors in the project completion; the *explanatory capability* of an algorithm suggests whether the explanation for a given classification decision can be easily understood from the classifier output. In (Alpoydin, 2004), it is argued that good classifiers should be able to predict the correct output for new instances after learning from the training set. In other words, they should have the ability to *generalise* well from the training data. Other desirable characteristic of a good classifier is its *robustness*: the ability to generate good results despite the existence of anomalies in the data caused by noisy data due to incorrect labels, imprecision in data collection and measurement errors. In the next section, a survey of classification algorithms is presented, discussing them in the context of their strengths and weaknesses in practical applications and in text mining in particular.

### 3.3.3. Nearest Neighbour Methods

Nearest neighbour methods are considered one of the simplest and most yet effective classes of classification algorithms in use. Their principle is based on the assumption that, for a given set of instances in a training set, the class of a new yet unseen occurrence is likely to be that of the majority of its closest "neighbour" instances from the training set. Thus the *k-Nearest Neighbour* algorithm works by inspecting the *k* closest instances in the data set to a new occurrence that needs to be classified, and making a prediction based on what class the majority of the *k* neighbours belong to. The notion of closeness is formally given by a distance function between two points in the attribute space, specified a priori as a parameter to the algorithm. An example of distance function typically used is the standard Euclidean distance between two points in an *n*-dimensional space, where *n* is the number of attributes in the data set.

When using nearest neighbour methods the clear decision to be made by the data miner relates to the choice of values for k and distance function that optimize results for a given training set. Choosing too small a value for k, for instance, *k=1* neighbour may cause the algorithm to become much too sensitive to training data and thus instable when new occurrences are inspected. On the other hand, large values of k may cause the algorithm to include much too distant points, which are not necessarily close to the instance being inspected, and algorithm may loose some of its predictive power. There are theoretical results indicating that *k* should grow as the number of instances available for training grows (Kulkarni et al, 1998), other results however indicate that for sufficient large training sets there is little benefit in increasing k beyond very small values (Cover et al, 1967). With respect to distance functions, the best choice will depend on the data being used. Euclidean distance is a popular choice with numerical data, and often extended to incorporate weights representing a measure of importance of each attribute (Hand et al, 2001). In text mining, document similarity by word co-occurrence is commonly used as a distance function (Weiss et al, 2005)

Nearest neighbour methods have the advantage of being simple to understand and implement, and there is sound theoretical foundation for the convergence of nearest neighbour method to the best possible solution as the training set increases (Cover et al, 1968; Kulkarni, 1998) . Other important factors are its ability to easily handle missing values (by selectively eliminating from the distance calculation the dimensions where values are missing) and ability to easily implement a reject option when the confidence of a prediction is not acceptable (Hand et al, 2001). Classification based on distance functions often translate to a certain measure of similarity between data points, which facilitates the interpretation of results by simple comparison to its closest neighbours. The method however has certain drawbacks: numerical values are required for the calculation of distance metrics between data points, and thus the algorithm does not readily work with categorical data; also, because it is based on a distance metric, it can be sensitive to dimensions with comparatively larger values, a problem that can be mitigated by introducing weights to dimensions in distance calculations or adjusting values as a data pre-processing step. Also, nearest neighbour methods tend to perform poorly on high dimensional attribute spaces with limited training data, as distance values used for prediction decisions get exacerbated with the growth in number of dimensions in the data set (Hand et al, 2001). In fact it can be shown that for nearest

neighbour methods, the size of the training set required to achieve acceptable levels of performance increases exponentially with the dimension of the data set (Kulkarni, 1998). The method also contains certain scalability issues, caused by the fact it needs to store all data points from the training set in order to calculate distances and compute a prediction.

### 3.3.4. Tree Based Methods

Tree based methods attempt to solve the classification problem by recursively finding partitions of the solution space based on data attributes which optimally separate the classes being predicted. This procedure is performed recursively on the training data set, so that each stage introduces a decision point related to an attribute. In the data set example from table 4, for instance, a decision rule could be created on attribute "Employment Status", partitioning the data according into "Unemployed" and "Employed, Self-Employed" when classifying on the target variable "Loan Approved". From this rule, further rules could be recursively devised for other attributes such as "Home Owner", "Salary", etc. The end result is thus a tree where each tree node creates a decision rule for classifying the data based on a given attribute. The process of creating new nodes (e.g. new branches in the tree) stops when a given threshold such as tree size or number of instances found for a given node is reached.

The key issue to consider when designing a tree-based algorithm is how to choose a data attribute to be used in the tree node such that the solution space is optimally divided according to the class being predicted. To this end, attributes can be ranked according to a scoring function that represents the greatest possible improvement on the tree classification. One approach would be to simply apply the loss function, or classification error described in 3.3.2 (Hand et al, 2001). However the development of tree-based algorithms has seen other scoring methods being proposed providing more effective results (Kulkarni et al, 1998). For instance, the ID3 algorithm (Quinlan, 1986) uses a scoring function based on the concept of information gain, and attribute selection is based on minimising a measure of entropy on the attributes being investigated.

Decision tree methods have the advantage of providing an output model that is relatively easy to interpret and explain: to understand how the classifier arrived at a particular decision, one must simply retrace its steps from the root of the tree, following decisions taken at each node. This class of algorithms also have the ability to handle both numerical and categorical data easily, and achieve relatively quick execution times: once the tree model is built, finding a classification decision for a new instance requires simply traversing the decision tree. Decision trees however have the potential downside of long training time, since data needs to be partitioned on a given attribute at each node and the number of nodes doubling each time the decision tree grows by one level, although algorithmic improvements are available to mitigate this type of problem. Another consideration is the fact that decision trees are a *monothetic* class of algorithm: because each node is split according to one attribute, only one attribute is considered at a time. This has the potential performance downside on problems where data is better described (and partitioned) as a combination of more than one attribute, such as a in a linear combination of attributes (Hand et al, 2001).

### 3.3.5. Naïve Bayes

The Naïve Bayes classifier uses a probabilistic approach for predicting the class of a given data point. The starting point is the Bayes theorem for conditional probability, stating that, for a given data point $x$ and class $C$:

$$P(C/x) = \frac{P(x/C) \cdot P(C)}{P(x)}$$

Furthermore, by making the assumption that for a data point $x = \{x_1, x_2, \ldots x_j\}$, the probability of each of its attributes occurring in a given class is independent, we can estimate the probability of $x$ as follows (Hand et al, 2001):

$$P(C/x) = P(C) \cdot \prod P(x_j/C)$$

Training a Naïve Bayes classifier therefore requires calculating the conditional probabilities of each attributes occurring on the predicted classes, which can be estimated from the training data set. Naïve Bayes classifiers often provide good results, and benefit from the easy probabilistic interpretation of results. However, the model's

main weakness lies on the assumption of independence of occurrence of attributes. This may not always hold on real data sets, where attributes may be strongly correlated. Consider, the data set presented in Table 5, where the “salary” and “home value” attributes, for instance, can be expected to be correlated.

### 3.3.6. Large Margin Classifiers: Support Vector Machines

Support Vector Machines are a class of algorithms for classification that belong to parametric methods – that is, finding an adequate function that partitions the solution space so as to separate the training data points according to the class labels being predicted, under the assumption that future prediction follows the same pattern. In the simple case where a linear function divides the two classes, a resulting hyperplane partitions the solution space. The following graph illustrates dividing hyperplanes for a sample of points belonging to 2 classes:

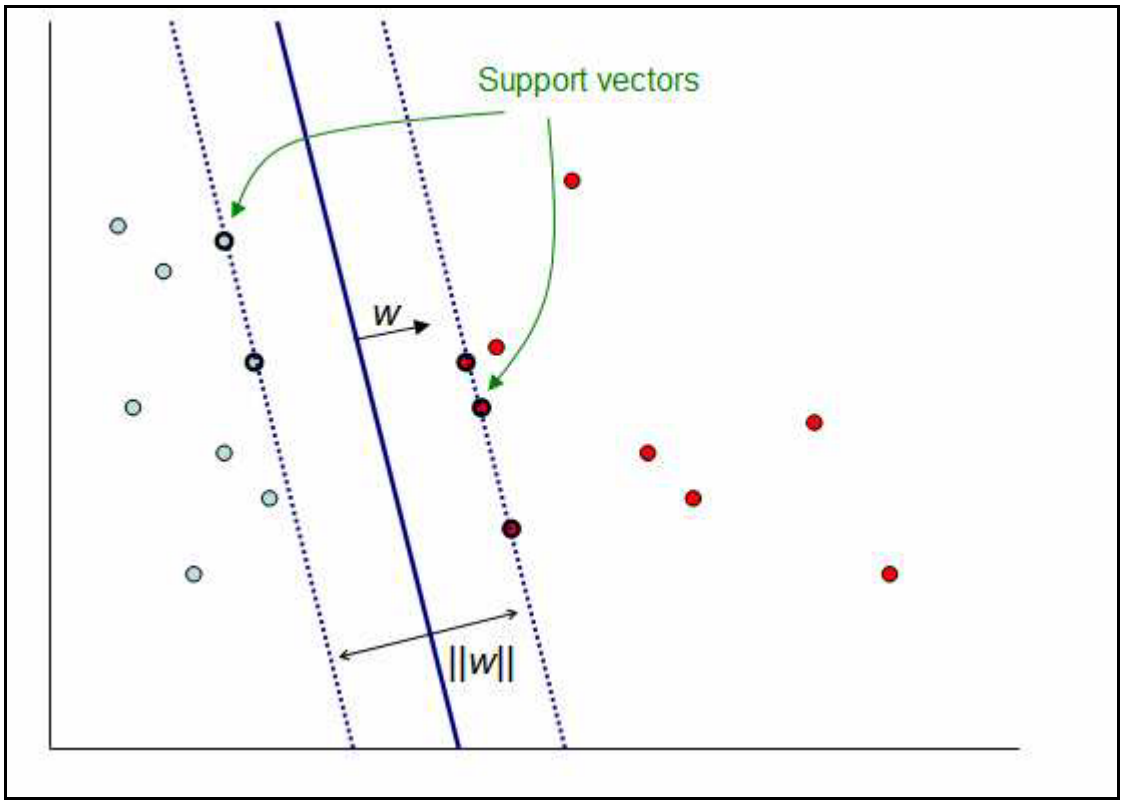



**Figure 8 - Hyperplane Separating Two Classes**

In the above example, there is a potentially unlimited number of separating hyperplanes dividing the two classes. In choosing the best possible one, an intuitive idea would be to choose a hyperplane that has the largest distance between any points from either class, thus creating the widest possible *margin* between points from the

classes. The intuition behind this method is that a hyperplane with a large margin would be a "safer" classification boundary, less likely to make prediction errors by being to close to the boundaries of one of the classes. Finding such hyperplane is the objective of the *Support Vector Machine* algorithm, first presented in (Boser et al, 1992). To achieve this, the problem can be formalised as finding the vector *w*, such that:

For classes $C_1$ and $C_2$, and feature data vectors $X = \{ x^t, r^t \}$ where:

$$r^t = +1 \text{ if } x^t \in C_1, \text{ and}$$

$$r^t = -1 \text{ if } x^t \in C_2$$

Find $\boldsymbol{w}$ and constant $w_0$, such that the dot product of *w* and a given data vector is as follows:

$$w^t \bullet x^t + w_0 \geq +1 \text{ if } x^t \in C_1, \text{ and}$$

$$w^t \bullet x^t + w_0 \geq -1 \text{ if } x^t \in C_2$$

And $\boldsymbol{w}$ has maximum length (Alpoydin, 2004).

The above equations state that points belonging to class $C_1$ and $C_2$ are on separate sides of the orthogonal hyperplane defined by the vector $\boldsymbol{w}$, and by maximising the vector length $||w||$, we obtain a dividing hyperplane with maximum distance between points from either class. It is demonstrated in (Boser et al, 1992) that finding this optimal hyperplane translates to a quadratic optimization problem, and whose complexity depends on the number of training vectors N, but not on the dimensionality of the data set. This is an interesting feature of this method as it has the potential to address the requirements of high dimensionality data sets, and as seen in (Alpoydin, 2004), the time complexity of this method has an upper bound of $N^3$. Another interesting result is that the model obtained from training support vector machines takes into account only the data points close to the dividing hyperplane for predictions: these are called the *support vectors*, and are expected to be in much smaller number than the entire data set, thus providing an algorithm with good performance during execution time.

It can not always be assumed that the classes can be suitably divided by a linear hyperplane, and in some data sets this assumption may be voided by noisy data, or by the nature of the data sets. On those cases, the method provides the ability to introduce an error constant during the learning process, where a penalty score C is added to points too close, or beyond the dividing hyperplane, thus allowing for some flexibility on misclassified points. Another important feature of Support Vector Machines in dealing with cases not linearly separable is its ability to map the problem space into another, possibly more convenient space by means of a *kernel function*, where the points allow for better separation. Several kernel functions have been employed to support vector machine classification, and have been surveyed in (Alpoydin, 2004).

As noted in (Burges, 1998) the theory of Support Vector Machines does not guarantee high performance of the method on all cases, however one interesting result as noted in (Alpoydin, 2004; Burges, 1998) states it can be demonstrated that the expected error for a Support Vector Machine classifier is a function of the number of support vectors, and not its dimensionality. This upper bound could translate to good performance, especially on high dimensional data sets. Also, some results from statistical learning theory suggest a relationship between large margins obtained by the method and reduced upper bounds on classification error (Kulkarni et al, 1998). In any case, the method has reportedly performed very well on empirical experiments, ranging from image recognition (Boser et al, 1992), text classification (Joachims, 1998), to opinion mining (Pang et al, 2002; Kennedy et al, 2006; Pang et al, 2004).

### 3.3.7. Considerations on Classifier Techniques

In the sections 3.3.2 to 3.3.6 of this chapter, common classes of classification algorithms were surveyed, with emphasis on their underlying motivation, applicability and positive aspects. It can be seen from the algorithms inspected that each implements a specific heuristic to address the lack of information regarding the unknown real distribution of data. Therefore, each method makes assumptions on how the predicted classes can be separated: for the Naïve Bayes algorithm, this corresponds to its reliance on the probabilistic independence of attribute occurrence, and in Nearest Neighbour methods, its assumption that data belonging to the same class are close by a certain similarity measure. These assumptions reflect the *inductive bias* of a certain algorithm,

and should be understood so that performance results can be correctly interpreted (Alpoydin, 2004).

To summarise the findings of this survey the results on each method's assessment is presented in the table below.

| **Algorithm** | **Positive Aspects** | **Negative Aspects** |
|---|---|---|
| k-Nearest Neighbour | Simple to understand and implement.<br>Easy interpretation. | Potentially slow as training data increases.<br>Categorical or missing values need to be pre-processed. |
| Decision Trees | Model is easy to interpret.<br>Handles both numerical and categorical data. | *Monothetic*, potentially leading to sub-optimal solutions.<br>High training times. |
| Naïve Bayes | Model is easy to interpret.<br>Efficient computation. | Assumption of attributes being independent not necessarily valid. |
| Support Vector Machines | Very good performance on experimental results.<br>Low dependency on data set dimensionality. | Categorical or missing values need to be pre-processed.<br>Difficult interpretation of resulting model. |

**Table 6 - Survey of Classification Methods**

As noted earlier in this chapter, the above survey aims at presenting popular classification techniques and their approaches to data-driven prediction, and represents only a fraction of available classifier techniques, with many more constantly being developed. Other methods based on the same principle present on Support Vector Machines, of finding a dividing hyperplane can be seen in the *linear discriminant* class of methods presented in (Hand et al, 2001; Weiss et al, 2005). A closely related method that received much attention on pattern recognition problems is Neural Networks, presented in (Kulkarni et al, 1998; Geman et al, 1992). A variety of

methods applied to the classification of textual data is surveyed in (Sebastiani et al, 2002).

**Choosing a Classifier**

The analysis of some of the available classification methods provided in this section highlighted key characteristics to observe when choosing a classifier for a particular task. The nature of the data set should be taken into account, in terms of dimensionality, size and data characteristics. Most methods presented handle only numerical data, and categorical or missing values need to be addressed before training. Training and runtime characteristics of each algorithm must also be taken into account, as time requirements are a determining factors to project success, cost and ultimately to the usefulness of the data mining task. On some cases, the high dimensionality or sheer size of the data set may forbid the application of slow performing methods. The explanatory capabilities of the method also need to be taken into account, and may constitute a key factor in the choice of data mining algorithm, depending on the expected outcomes of the data mining exercise by the end users.

Finally, it would be ideal to choose the classifier with the best performance in terms of minimisation of classification error. In other words, a classifier that makes as little mistakes as possible on its predictions. Determining which classifier will perform best is dependant on a number of factors related to the availability of data, such as distribution of the class label in the total population - an unknown fact in principle - and how closely that distribution is represented in the data attributes available for training. It is desired that the distribution present on the training set closely reflects that of the entire population but this however can not always be guaranteed. The size of the training *set* is also relevant, since larger training sets tend to approximate the performance of a classifier to the best obtainable performance in its class, as theoretical results for various methods demonstrate (Kulkarni et al, 1998). In addition, induction bias has a part to play in choosing an algorithm since a particular heuristic present in one technique may better discriminate classes than other methods for a particular problem.

In general, however the available data is the single most important source of information and intuition on deciding on a classification method, and the

recommended approach in the literature for choosing the best performing classifier for a given problem is not based on algorithm principles or theoretical results, but instead it advocates proceeding in a *data driven* approach, by making the best use of the available training set and experimenting with different methods and parameters (Hand et al, 2001; Kulkarni et al, 1998; Weiss et al, 1998).

### 3.3.8. Evaluating Classifier Performance

Determining how well a classifier will make predictions on unseen data is one of the most crucial aspects of any supervised learning task. A series of methods for evaluating classification performance are investigated in this section.

**Testing on Unseen Data**

If the performance of a classifier is tested against data used for the training process, it can be expected that the predictions will be optimistically biased, since these are data points already "seen" by the classifier (Hand et al, 2001). Thus, a better option is to test the classification results on a separate data set, not used during training, and the data set can be divided into two sections: the *training* set is a subset of data used to train the classifier algorithm; and the *validation* set is used to evaluate predictions using the trained algorithm, and measure classification results. This strategy will allow the classifier to be tested on data points not used during training, and therefore yet unseen by the classifier.

A further extension of this approach takes into account the fact that when testing only on a particular subset of data, there is a chance of the algorithm performing unusually good or bad simply by chance, as a result of the selected data points for each of the subsets. To mitigate this problem, the training and testing cycles can be repeated using different subsets from the data set, in a process called *cross-validation*. The idea of cross-validation is to subdivide the data set into various subsets, or *folds*, to be used as the test set, while the remainder of the data set is applied for algorithm training. A 10-fold cross validation will generate 10 training and testing cycles, thus evaluating the performance of the algorithm when different sets of data are used for training. Upon each cycle, algorithm performance can be measured using adequate metrics, and inspected individually, or averaged over all folds.

It is suggested in (Hand et al, 2001), that while cross-validation is a robust and often used method, the validation set, though not used in training, becomes part of the design process of the classifier. To avoid further bias on the validation set, a third *test set*, or "hold out" set is advocated to be used purely for results confirmation, for the cases where enough training data is available. The use of data sets from different time periods is suggested in (Weiss et al, 1998), in order to avoid any temporal bias on data sets where time is a factor.

**Performance Metrics**

As formalised in Section 3.3.2, classification performance is usually measured by a loss or error function over prediction results. The choice of error function will also drive the iterative improvement of classification parameters for a certain algorithm, and should be consistent with the requirements of the data mining task. The *classification error rate* or *misclassification rate* amounts to the proportion of classifier predictions that are incorrect, and is given by the formula, for a data set of total size *N*:

$$Error\ Rate = \frac{Classification\ Errors}{N}$$

In many cases, a classifier may show low error rates but still display undesirable classification behaviour. If, for instance, a loan applications data set with a very high percentage of negative cases is provided, the classifier may choose to simply make negative predictions for every new case seen, which would still generate low error rates, but its results would be of little practical use. To better illustrate the possible types of classification error, results are often displayed in terms of correct and incorrect classifications per each class, in a *confusion matrix* (Weiss et al, 1998) as shown below for a classification problem with two classes (positive and negative).

| **Predicted Value** | **Real Value** | |
|---|---|---|
| | Positive | Negative |
| Positive | True Positive | False Positive |
| Negative | False Negative | True Negative |

**Table 7 - Confusion Matrix for 2-Class Classification Problem**

To ensure the classifier is in fact detecting the correct classes, and covering a suitable number of cases on each class, the notions of *precision* and *recall* can be used. These are given by the formulas below, as presented on (Weiss et al, 2005):

$$Precision = \frac{Correct\ Predictions\ for\ Class}{Total\ Predictions\ for\ Class}$$

$$Recall = \frac{Correct\ Predictions\ for\ Class}{Total\ Entries\ for\ Class}$$

Precision indicates the rate at which a classifier makes a correct prediction, or the percentage for which its predictions are correct. A high-precision classifier on the positive class would have high true positives and low false positives. Recall relates to how many predictions for a given class are made, out of the total available cases for that class. A high recall classifier would have high true positives with low false negatives thus covering all entries labelled "positive". The above formulas can be rewritten using the above confusion matrix, and are shown below for the "positive" class:

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives}$$

$$Recall = \frac{True\ Positives}{True\ Positives + False\ Negatives}$$

There is an inherent trade-off between precision and recall: by increasing the precision of a classifier, it is made more specific and thus more "conservative" in making a prediction, thus lowering recall. On the other hand a high recall classifier might be tuned to make predictions more "generously", at the expense of precision. Using the loan application data set as an example, a high-precision, low-recall loan applications classifier would make few loan approvals, but its decisions would be correct most of

the time. A low-precision, high-recall loan applications classifier would make incorrect predictions more often, but is more likely to detect a positive loan application.

**Accuracy and F-Measure**

Aggregated metrics incorporating precision and recall information are sometimes used to report research results. The *accuracy* refers to the overall classifier precision across all classes, and is given by the formula:

$$Accuracy = \frac{True\ Positives + True\ Negatives}{True\ Positives + False\ Positives + True\ Negatives + False\ Negatives}$$

In other words: the rate of correct predictions over all predictions. Accuracy is 1 when no classification errors are reported by the classifier. Accuracy however suffers from the same issues seen on misclassification rates, where a data set where a class contains many more occurrences than another can generate biased results. To mitigate this problem, the harmonic mean of precision and recall, or *F-measure* is often used:

$$F - measure = \frac{2}{\frac{1}{precision} + \frac{1}{recall}}$$

The choice of performance metric should take into account the data set and nature of the prediction problem being investigated: classification precision might be of more importance than recall, for instance on diagnostic systems with a high risk of misclassified occurrences might prefer high precision classification at the expense of recall. In the literature, F-measure is a common metric for reporting classification performance results, as seen on (Sebastiani, 2002), with accuracy often reported for the cases where the data set is has a balanced number of entries for positive and negative classes, as seen in (Pang et al, 2002; Pang et al, 2004).

### 3.3.9. Challenges to Classification in Data Mining

In this section some of the challenges and limitations inherent to the problem of supervised learning are explored in more details.

**Bias / Variance Trade-off**

Classification error can be seen as composed of two factors: one comes from the model complexity of a classifier, which dictates how much information can a classifier capture, and hence represent the data set nuances as closely as possible. The other indicates how much do classification results vary in terms of classification error when yet unseen data samples are presented for classification. These two aspects of a classifier are commonly referred to as *bias* and *variance*. The overall error for a classifier can be interpreted as the sum presented in the formula (Weiss et al, 1998):

$$Error = Optimal + Bias + Variance$$

Here, optimal error is the best possible classification error an ideal classifier can obtain given the data set in question, which can be non-zero, depending on the nature of the data, and how closely it represents the events being captured. To further reduce the error of a classifier, one could attempt to build models with increasing complexity, making it capable of capturing more information from the data. However, in doing so, a more complex model is also more likely to represent a particular training set very well, but not the underlying patterns from the data in general: the classifier's ability to generalise is reduced by *overfitting* the model to the training data. This state of affairs is what is commonly known as the *bias/variance trade-off*: attempting to reduce bias error by building more complex model generates classifiers less able to generalise well, thus increasing the variance component of the error (Hand et al, 2001; Kulkarni et al, 1998).

**Dimensionality**

Intuitively, it would appear sensible to progressively add features representing different aspects of the data to train a classification algorithm, hoping that this new information will assist in the algorithm's predictions. However, adding features is done at the expense of a requirement for more training data, and performance penalties in algorithm training and execution. This drawback is what is commonly called the *curse of dimensionality*. Adding an extra feature to the training data set of a classifier can be seen as adding another dimension on the search space the classifier now needs to inspect. This implies a larger effort required by training algorithms for searching

through the solution space for patterns. This can be seen for instance, on decision tree algorithms, where a decision tree needs to grow one level to accommodate a new feature, thus doubling the size of a tree which includes all attributes. Similarly, a nearest neighbour algorithm now requires the calculation of distance functions for an additional dimension across all of its training data set in order to make a prediction, thus affecting the runtime of the algorithm.

A larger search space not only affects algorithm performance: by increasing the size of the solution space, data points become more "distant" from one another. This can be noted in the example where a data set increases from a 2-dimensional plane to the 3-dimensional space. The calculation of distance between points now have an additional dimension to be added, as seen on the formula for Euclidean distance for both cases:

$$D_{2\text{dim}} = \sqrt{(a_x - b_x)^2 + (a_y - b_y)^2}$$

$$D_{3\text{dim}} = \sqrt{(a_x - b_x)^2 + (a_y - b_y)^2 + (a_z - b_z)^2}$$

The distance between data points can be expected to increase, or at the very minimum stay the same, as more dimensions are added, and could mean patterns become less pronounced for the same amount of training data. The sparseness effect on data points translates to a requirement for more training data to be available for a training algorithm to achieve similar performances. This is demonstrated in (Kulkarni et al, 1998) for various classes of algorithms, where the convergence to an optimal result as a function of training data available happens at a slower pace as the number of dimensions increase.

The concerns associated with the curse of dimensionality have created the need for strategies that can reduce the number of features while improving or preserving classification performance as much as possible. Thus, a number of feature selection and reduction techniques have been proposed in the literature. The more common approaches attempt to identify attributes with high correlation with the predicted

classes using statistical tests, or a measure of attribute information gain (Hand et al, 2001; Yang et al, 1997). Another approach known as principal component analysis involves merging several attributes together by using a linear function, where each attribute receives a weight according to its relative importance to prediction (Weiss et al, 1998). Feature selection methods have shown to perform similar to full-feature classifiers but using a smaller subset of attributes based on high-dimensionality text data sets, as seen in (Yang et al, 1997; Rogati et al, 2002; Forman et al, 2002)

This same concern has motivated the study of methods that achieve better results on high dimensional data sets, such as textual data sets, as will be shown on the next chapter have this characteristic, containing usually several thousand attributes. Support Vector Machines are a method whose training time is dependant on the size of the training data set, but not on its dimensionality (Alpoydin, 2004), making it a good candidate for this type of data sets. Another compelling feature of Support Vector Machines is that the method has an expected classification error dependant on the number of support vectors but not data set dimension, thus making it a suitable technique for addressing certain dimensionality issues (Burges, 1998). Indeed, Support Vector Machines have shown very good empirical results when applied to the problem of high dimensional text classification (Joachims, 1998).

**No Free Lunch**

It was noted in the previous Section, that a wide choice of classification methods is available to the data miner for performing a prediction task, with new approaches constantly being developed. It would be desirable to know if there is a classification algorithm that can consistently provide better classification performance than the others, or if such algorithm could exist in theory. In the mid 1990s however, a result presented in (Wolpert, 1996) has demonstrated this is not the case: The *No Free Lunch* theorem for classification states that, assuming no prior knowledge of the classification problem is present, all classes of classification algorithms will obtain the same classification performance *when averaged over all possible learning scenarios*, and this performance will be no better than random guessing, on average. In other words, there is little hope of devising a superior generic classifier algorithm that can consistently outperform all the others on all scenarios.

In practical terms what this impressive result suggests is that good results obtained by a specific classifier to a specific data set reflect the "fitness" of the algorithm's heuristics to one particular case, which is not an indication the algorithm will perform any better on a different scenario. This result reinforces the view that choosing what classifier to apply on a specific problem should be done in a data driven fashion, validating performance results against other models where possible, as discussed on Section 3.3.7 and seen in (Hand et al, 2001; Kulkarni et al, 1998; Weiss et al, 1998).

The No Free Lunch theorem also has implications for cross-validation techniques, stating that cases where an algorithm obtains better results in cross-validation are no indication the method will perform equally well on unseen examples. However, as seen in (Wolpert et al, 2001) this result does not imply cross-validation is not a valid and workable technique, but it shows the reliability of this approach can not be formally justified for all cases, and should be used with caution and within certain assumptions. Despite this caveat, cross-validation is widely regarded as a good method for testing and reporting on classification performance in the literature.

## *3.4. Data Mining Tools*

This section discusses software applications for performing data mining tasks, investigating commercial and open source offerings, from off-the-shelf packages to industry specific products. To illustrate the typical capabilities of a data mining package the open source *RapidMiner* package is presented in more details.

To illustrate the evolution of commercial data mining applications, three stages of development were proposed in (Piatetsky-Shapiro, 1999): the **first generation** of data mining applications appeared in the 1980's and comprised mostly of tools originated by research and dedicated to a single task, or single algorithm, such as performing decision tree classification, or clustering. This class of tools required technically savvy users and a thorough understanding of data mining techniques. Integration between tools and between tools and other data applications was non-existent. Later, as data mining found its way to wider commercial uses, software vendors in the mid 1990s started offering the **second generation** of data mining suites that would cover not only a larger set of data mining techniques, but also would support other important activities

of the knowledge discovery process such as data pre-processing and visualisation. By this time a survey conducted in (Goebel et al, 1999) found as many as 37 products aimed at performing one or more data mining activities and having achieved a sufficient level of maturity to warrant their use as commercial applications. The task of data mining however remained highly technical, and far too complex for being directly applied by business users. Hence, the **third generation** of data mining tools incorporates domain specific vertical knowledge in a given industry, integrates with company's legacy systems and provides useful, intuitive interfaces for business users aiming at solving a specific business problem such as email spam detection, fraud detection, customer analysis and telephony network analysis.

### 3.4.1. Open Source Tools and RapidMiner

In the open source area, popular research applications have gained popularity with practitioners. The wealth of options for open source data mining can be seen on the list of available packages in (KDNUGGETS, 2009) and the Machine Learning Open Source Software portal and conferences, providing a centralised forum for open source data mining (Sonnenburg et al, 2009). One important contribution to the open source data mining community is the the *Weka Toolkit* (Witten et al, 1999), developed at New Zealand's University of Waikato, a popular data mining package available under open source license: it is a comprehensive suite of Java class libraries for performing a number of data mining tasks. Weka algorithms can be accessed programmatically from another application, or directly via its user interface and is now maintained by the open source community.

**RapidMiner**

RapidMiner is an open source data mining suite also available under a commercial license. It emerged from the *YALE* data mining environment (Mierswa et al, 2006) originally designed to be a rapid prototyping system where data mining implementations could undergo a proof-of-concept using a tool that can easily build, execute and validate data mining models, before the need to develop a more complex solution. RapidMiner has evolved into an offering with commercial strength features such as:

- Ability to quickly prototype data mining tasks on a graphical user interface.

- Developed in Java and platform-independent.
- Support for a wide range of tasks on data analysis, prediction, clustering and visualisation.
- Additional functionality specific to text mining
- Integration with algorithms implemented for the Weka toolkit, making them accessible from inside RapidMiner.
- Integration with relational database systems via JDBC interfaces.

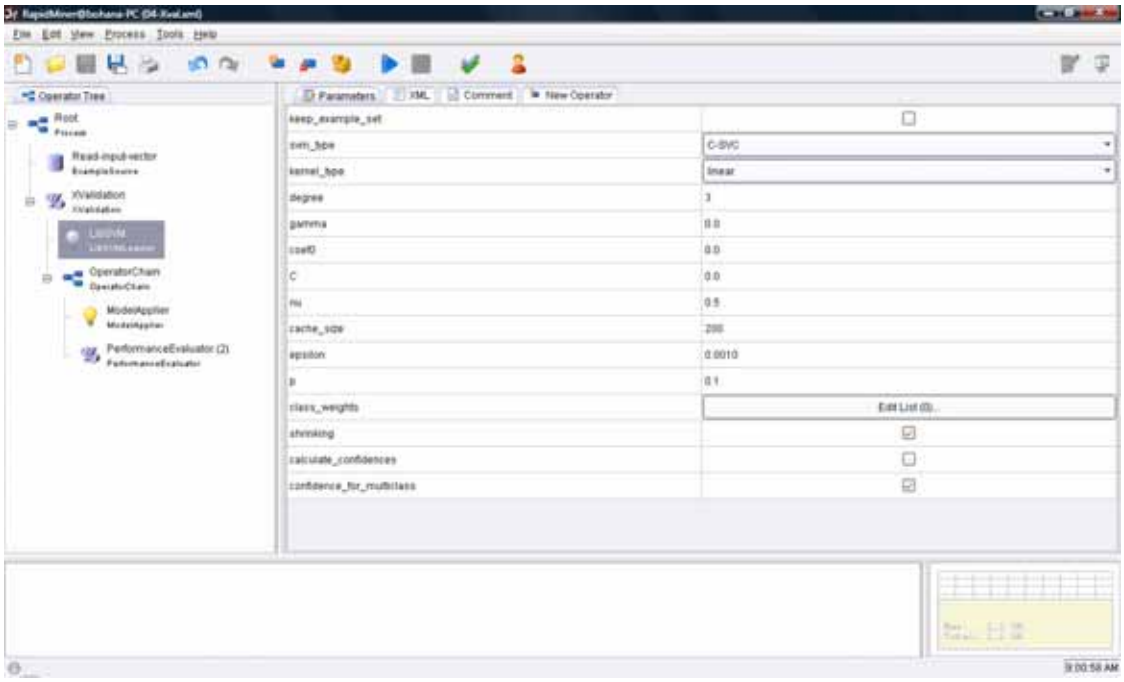


**Figure 9 - RapidMiner WorkBench**

## *3.5. Conclusion*

This chapter presented a discussion on how to perform data mining, what are the key objectives of a data mining task, and how to go about executing them in a systematic way. The area of knowledge discovery processes was introduced, highlighting key methodologies available from both the industry and research. Knowledge discovery processes embed knowledge and expertise obtained from practitioners and lessons learned from previous projects that can be reused in future projects, such as the data mining experiment performed as part of this dissertation. The KDD Process and CRISP-DM methodology were presented in more details, and important aspects

present on these methodologies were discussed, such as its iterative nature, the importance of data preparation and choice of data mining tasks to be performed. Furthermore, the link between knowledge discovery processes and knowledge creation, and its implications to knowledge management were also investigated.

The key activities of data mining process were surveyed, presenting their objectives and examples where each particular activity had a successful application, leading to a more focused study on supervised learning and classification – the scope of this dissertation. This chapter surveyed different classes of classification algorithms, and discussed aspects to be taken into account when selecting a classifier for a prediction task, such as data characteristics, algorithm performance and explanatory capabilities. Common metrics for measuring classification performance used in the literature were surveyed, and the challenges and limitations of supervised learning were discussed: the curse of dimensionality and how it affects a model's ability to produce good results in a timely fashion with limited training data; it was also seen how the bias/variance trade-off imposes limitations to the complexity of an algorithm at the expense of classification performance. Finally, the No Free Lunch theorem demonstrates the impossibility to obtain a classifier that can consistently outperform all other methods, on average, and how this affects the choice of a classifier to a specific problem. Finally, the chapter concluded with a discussion on available data mining tools and features, to be taken into consideration for the implementation aspects of this dissertation's experiment.

The main outcomes of the review of the data mining literature presented on chapter's was to survey and identify important aspects from knowledge discovery processes, how they can be used in this dissertation's experiment, and their relationship to knowledge creation and knowledge management; to further understand data mining tasks, and survey the state of the art in classification algorithms, performance measurements and limitations inherent to the problem of data classification; and finally to evaluate the state of the art in data mining applications, identifying potential candidates to be used as part of a data mining experiment. In the next chapter the areas of text mining and opinion mining are presented, and the SentiWordNet lexical resource is discussed concluding the review of the literature in preparation for the experiment's setup, execution and results evaluation.

# 4. TEXT MINING AND OPINION MINING

This chapter reviews research literature in the fields of text mining and opinion mining. It discusses the motivations for performing knowledge discovery on text data sources, and illustrates how the field of text mining is closely related to that of data mining in general, but with its own additional research concerns stemming from the need to understand and process the complexities and nuances of unstructured text data. The relevance of text mining technology to the field of knowledge management is also explored.

Next, the field of Opinion mining in text is discussed. Opinion mining is a relatively new and challenging field dedicated to detecting subjective content in text documents, with a variety of uses in real world applications. It is also the main subject of this dissertation's experiment, thus a thorough survey on the state of the art in approaches to performing opinion mining tasks is presented, and the research is placed in the context of the objectives of this research.

## *4.1. Text Mining*

The key driver for exploring text data from a knowledge discovery perspective, as is the case with knowledge discovery in databases, is the abundance of available textual data in digitised format. Text is a rich and natural means of storing and transferring information, with the Internet being one of the most notable examples: it has been estimated that over 3 billion available documents in textual format have been indexed by the most popular Internet search engines (Sullivan, 2005). The situation is similar inside organisations, where a large variety of textual data within emails, memos, wikis, portal pages and corporate documents are now fully authored and made available in digital format, with some estimates indicating that up to 85% of corporate data is stored in the form of unstructured text documents (McKnight, 2005). The availability of information in text format suggests an opportunity for improving corporate decision making by tapping on text data sources, and the large volumes are thus likely targets for automated methods of discovering new knowledge. This opportunity triggered the development of the emergent area of knowledge discovery in texts, or text data mining (Feldman et al, 1995).

The fundamental implication of using text data sources for performing knowledge discovery is that data is *unstructured* in nature: as postulated in (Weiss et al, 2005) there are no special requirements for including a text document in a text mining task as there would be on a typical data set where the data is already organised with clear semantic meaning, using pre-defined data types, labels and ranges of values. Text documents on the other hand are far more flexible and richer in their expressive power, but such benefits are constrained by the added complexity inherent to the vagueness, uncertainty and fuzziness present in any natural language (Hotho et al, 2005). For this reason, the discipline of text mining or knowledge discovery in text leverages contributions from different research areas in computer science, such as computational linguistics, artificial intelligence, information retrieval and machine learning (Herschel et al, 2005). The definition of what is considered within the realm of text mining varies and sometimes overlaps with that of other disciplines that are also concerned with the computational treatment of text data, such as information retrieval and natural language processing (Kroeze et al, 2003).

In (Weiss et al, 2005) a task oriented view of text mining is proposed, encompassing the following aspects:

- **Information Retrieval** or obtaining a subset of documents from a document corpus based on user specified search criteria, as observed on internet search engines and document searching capabilities of text knowledge repositories.

- **Information Extraction**, which deals with extracting specific information from text documents, such as extracting the date and time an event occurred from news documents, or numeric values for a given attribute – the price of a given asset for example. Text summarisation techniques that aim at providing a condensed representation of the information contained in a document would also fall in this category.

- **Text Data Mining**, the application of data mining techniques on text data sources, such as classification, clustering and exploratory data analysis for the purposes of extracting new and useful information.

The above suggests text mining involves a wide range of techniques for the treatment of text, and whose objectives are not strictly restricted to the discovery of knowledge. A similar definition to the above is observed in (Hotho et al, 2005). However, in (Hearst, 1999), a clear distinction is made between information extraction and retrieval techniques and text data mining, arguing that for the purposes of information retrieval, the information contained in documents is already known (at least by the authors), and therefore would not fall within the scope of the discovery of new knowledge. According to Hearst, data mining on text concerns the use of "text metadata to tell us something about the world, outside the text collection itself", whereas text data mining involves the exploratory analysis of text documents for deriving new knowledge. In (Feldman et al, 1995) *Knowledge Discovery in Text* is described as the application of knowledge discovery methods to textual data, closely resembling that of text data mining seen above.

From the above discussion, the definitions of text mining in literature can be broadly classified in two types: first, text mining can be defined as all activities involving the treatment of text for analytical purposes, including extraction and retrieval techniques; second, text mining can be seen exclusively as text data mining in line with the objectives of the definition of knowledge discovery stated in (Fayyad, et al, 1996), thus leveraging text as the source of data for the discovery of new yet unknown knowledge. It is worth noting however that in any case, text data mining is closely linked to other research fields involving the computational treatment of text, and it is not uncommon to see the application of text mining techniques to related areas and vice versa: discovering new patterns in text may come into assistance to information retrieval, and information retrieval and knowledge extraction are useful techniques for text data mining, as will be further illustrated in the following sections discussing text mining applications and techniques.

### 4.1.1. Applications of Text Mining

**Exploratory Text Analysis**

The exploration of large text data sets is a useful approach for obtaining insights from data not usually possible by manual inspection. The value of this type of investigation to creating new knowledge can be seen in the breakthrough analysis of (Swanson et al, 1997), where mining relationships between research documents produced in distinct areas has led to the corroboration of new hypothesis in the medical domain. Many other approaches of successful systems and prototypes incorporating data mining methods for data description and visualisation are reported in the literature. In (Nasukawa et al, 2001) a system for the interactive analysis of patterns in text is presented, with a case study on support tickets where documents can be analysed by their correlation to categories, urgency and client feedback. In (Dorre et al, 1999) descriptive text mining techniques are applied to improve customer relationship management. Another innovative use of text sets is finding trends in documents creation according to topic, timeline or keywords. A prototype trend analysis system based on phrase similarity measures is demonstrated in (Lent et al, 1997) applied to the investigation of patents; another approach using measures of word co-occurrence and the assistance of a pre-defined taxonomy is investigated in (Feldman et al, 1998) for investigating trends and similarities amongst different news data sets.

Visualisation techniques for exploring documents clustered into topics, and graphs representing relationships between entities such as companies and executives are presented in (Feldman et al, 1998-b). An approach for organising documents into using clustering techniques based on a legal documents data set is presented in (Conrad et al, 2005). Other approaches to extracting information from document collections based on visualisation techniques are surveyed in (Hotho et al, 2005).

**Information Extraction**

Information extraction concerns the identification of relevant information present in text documents, which can be extracted into a more structured database for further use, or used as additional metadata in the exploration of text sources. Automated methods could be employed for example, to extract company, industry and executive names from news sources to build a searchable database of company details. Systems that perform information extraction have been applied in law enforcement to help in the analysis of seized documents, where entities such as name, addresses and bank accounts were extracted into a database for the purposes of visual analysis and reporting (Weiss et al, 2004); In (Ghani et al, 2006), product attributes were extracted from text sources to enrich the content of a decision support system based on transactional data, and had further uses in competitive intelligence and recommendation systems; In (Dorre et al, 1999) another example of attribute extraction from financial news is presented, for the purposes of exploration of documents.

**Automatic Text Classification**

Text classification involves the application of classification techniques in text data for the prediction of a class for a given document. One common use of text classification is in automatic text categorisation according to topics, as seen in the categorisation of news sources in (Joachims, 1996; Joachims 1998); a similar example can be seen in (Forman et al, 2006) for the automatic categorisation of incoming technical support requests per product type, reducing manual intervention and accelerating routing of the call to the correct support teams. Supervised text classification techniques are also at the core of many approaches for filtering unsolicited content such as email spam (Provost, 1999; Meyer et al, 2004; Kolcz et al, 2001). The classification of text for forensic purposes such as author identification has also been studied in (Corney et al 2002). Text classification techniques will be employed as part of this dissertation's experiment, and are discussed in more details in Section 4.1.3 of this chapter.

**Text Mining in Knowledge Management**

As shown in the above examples, text mining technology opens opportunities for the creation of new knowledge from text, enriching data sources with extracted data from textual documents and optimising information retrieval from repositories for decision support. The large collections of text data in digital format available in today's companies suggest those techniques are useful tools for knowledge management systems. Indeed, the application of data mining techniques to text, or text data mining are aimed at creating new knowledge and could therefore yield the same benefits to knowledge management, innovation and competitiveness studied in the Chapter 2.

Applications of text mining technology as an assistive technology for knowledge management initiatives are outlined in (Marwick, 2001), such as automated classification of documents into categories, the organisation of knowledge repositories, and text summarisation techniques to mitigate information overload. An example is the *eClassifier* application presented in (Cody et al, 2002) aiming at using text mining technologies for building taxonomies in explicit corporate knowledge repositories. Another approach is seen in (Kao et al, 2003) where a knowledge management system was developed for automatic organisation of knowledge hierarchies to facilitate the retrieval of relevant knowledge.

### 4.1.2. Representation of Text Data

To realise the benefits of text mining applications, strategies are needed to address the complexities and ambiguities of natural language. In addition, a structured representation of text that captures relevant information from documents is a necessary requirement for many text mining tasks such as classification and clustering. In this section the techniques for the treatment of natural language and approaches for representing text are surveyed.

**Natural Language**

Due to its lack of structure, text data normally undergoes a preparation stage that attempts to capture key components of natural language that will be employed on the text mining task. The treatment applied to the source data will dictate the model's characteristics and what information can be extracted from it. It is therefore important to match the preparation steps with the overall objectives of the exercise. The table below extracted from (Stavrianou, 2007) summarises the key concerns commonly found in processing natural language for data mining, with text preparation encompassing all but the last task.

| Issue | Objectives |
|---|---|
| Stop lists | Removal of terms occurring with high frequency and potentially of little relevance. |
| Stemming or Lemmatisation | Reducing words to a normalised form, or stem. |
| Noisy data | Correction of spelling mistakes, word shortenings and alternative forms. |
| Tagging | Adding syntactic categories to terms. |
| Word Sense Disambiguation | Determining meaning of ambiguous terms that best applies to context of text. |
| Collocations | Identifying terms represented by multiple words. |
| Tokenisation | Determine policy for grouping units of textual information. |
| Text Representation | Conversion of textual document into a model that best captures relevant features for text mining. |
| *Automated Learning (Text Data Mining)* | *Determining text mining approach.*<br>*Determining similarity measures.* |

**Table 8 - Key issues in Text Mining (Stavrianou, 2007)**

One of the first concerns in capturing relevant information from texts the creation of *stop lists*. These lists indicate what terms from the document collection are highly likely to appear, and carry little information when attempting to detect patterns. Common words in the English language such as "the", "and", "of" are usual candidates for stop lists. Care must be taken however to build a list with the specific

objectives of the data mining task in mind, since stop words may become relevant on different scenarios.

The objective of *lemmatisation and stemming* is to reduce the number of variations of a term by transforming similar occurrences to a canonical form, or lemma, or by reducing words to their inflection root, or stem. This will reduce the number of attributes to be analysed in the text collection, reducing noisy signals and the dimension of the data set. An example of stemming is the conversion of singular/plural and present tense/past tense into a single form. This pre-processing step is dependant on the objectives of text mining, and the loss of relevant information from stemming should be evaluated (Weiss et al, 2004). One widely used approach is the rule based method described in Porter's algorithm (Porter et al, 1980), whose implementation is now in public domain.

As with any data collected in uncontrolled environments, data clean up tasks to eliminate *noisy data* need to be taken into account. In text, data clean up issues take the form of spelling error and inconsistent spellings corrections, resolving term shortenings and abbreviations, stripping markup language tags, and converting text to lowercase or uppercase where appropriate.

There are instances in natural language where the same term may yield different meanings, depending on their use within the sentence, the domain the document belongs to. Consider for example the two very distinct meanings of the word "book" in the following sentences:

- *"This is a great book."*
- *"You can book your flights from this website."*

Discovering the meaning a specific term refers to in a sentence is known as *word sense disambiguation*, and is an active research topic in natural language processing and machine translation. This problem has received much attention in the field of machine translation from very early stages, where its intrinsic difficulties were noticed (Nirenburg, 1997): in broad terms, to obtain reasonable results in word sense disambiguation a large amount of information is required about the context the word is

being used – such as its role in the sentence and discourse aspects - and on the availability of external knowledge sources where sources of meaning can be queried. Several approaches for addressing word sense disambiguation are surveyed in (Ide et al, 1998), where it can be noted the use of external resources progressively evolved from manually building small disambiguation resources of restricted scope to the more recent data driven methods that leverage available knowledge resources in electronic format such as online ontologies, large annotated corpus and large scale manually derived lexical resources such as the WordNet database of term relationships (Miller et al, 1990).

*Tokenisation* refers to the process of segmenting a text input into its atomic components. The approach to tokenise a document depends on the mining objectives, one common approach is to use individual words as tokens, and spaces and punctuation marks as separators. However, punctuation marks can often be part of the analysis, as seen in (Corney et al 2002) and might instead be used as tokens. *Word collocations* are terms described by more than one word and should be referred to as a unit for analysis. Collocations can be found by statistical similarity when examining text sets, or derived via information extraction techniques and dictionaries.

Finally, depending on data mining objectives, it is important to determine the grammatical class a term belongs to. This can be done by attaching tags indicating the part of speech being used by a word in the text sentence. A *part of speech tagger* is an application that performs this task. Taggers are usually built by statistic analysis of patterns from large corpus of documents with annotated parts of speech, with The Penn Treebank (Marcus et al, 1993) and Brown Corpus (Garside, 1987) being popular examples. The Brill part of speech tagger (Brill, 1992) is one commonly used algorithm based on building tagging rules from annotated documents. Other approaches to part of speech tagging have been proposed using maximum entropy techniques (Toutanova et al, 2000) and in building statistical markov models (Brants, 2000). Several implementations of part of speech taggers can be found in the free NLTK toolkit for natural language processing (Loper et al, 2002).

The above discussion presented the main aspects of natural language processing that can be part of a text mining implementation. In the next section text representations for

mining text documents are discussed and techniques for text classification are surveyed in more details.

**Text Representation**

Before executing machine learning techniques on textual data, a structured document representation needs to be devised capturing as much statistical information about the document as possible. The most common representation formats are a variation of the concept of word vectors originally proposed in (Salton et al, 1975), and is based on the assumption a document can be characterised by the tokens, or words, it contains. Considering a document tokenisation based on words, at its simplest form a word vector data set similar to the model described in Chapter 3 can be built where each column represents information for a given word in the document, and each line represents a document in the collection. When a word is present in a given document, a non-zero value is present representing term presence. This can be, for instance, a binary value indicating a term has occurred in a given document. A partial word vector data set is represented in the figure below.

| ID | "ant" | "book" | "car" | "food" | … | "zen" |
|---|---|---|---|---|---|---|
| 1 | 0 | 1 | 0 | 0 | … | 1 |
| 2 | 1 | 0 | 0 | 0 | … | 0 |
| 3 | 0 | 0 | 0 | 1 | … | 0 |

**Figure 10 – Example Word Vector**

A common extension to this idea uses frequency information about a term, instead of a binary presence indicator. Attribute values are a numeric value indicating how often a term appears in a given document. The frequency information can be refined by balancing it with a measurement of the importance of a given term on the overall text collection, as high frequency terms that appear on most documents are less likely be statistically significant for pattern detection. In this case, a popular measure is the TF-IDF metric, or *term frequency – inverse document frequency* (Salton et al, 1987), given by the formula:

$$tf - idf\,(word) = tf\,(word) * \log\left(\frac{N}{df\,(word)}\right)$$

The first term is term frequency of a word in a document, and the second indicates the inverse document frequency of a word as a measure of how often the term appears in a document collection of size *N*, with *df* representing the frequency of a word in the document collection.

The number of columns in a word vector is a function of the number of distinct tokens in the document collection, or its *dictionary*. This number can grow quite quickly with larger and richer documents, and it is not uncommon to see very high dimensional word vector spaces with several thousand attributes. To mitigate the negative effects of high dimensionality, stop word lists and stemming are often used resources in reducing the final number of terms (Weiss et al, 2004; Sebastiani et al, 2001).

One natural extension of the above model is the use of more than one word to represent a column in the attribute vector. This approach can be useful to detect important collocations based on more than one term, sometimes referred to as *bigrams* (two word collocation), or *n-grams* for the generic case. Other combinations of single term unigrams, with multi-term n-grams are also possible, and could be effective depending on the domain and type of mining problem.

### 4.1.3. Document Classification Techniques

Of particular concern to this dissertation is the subject of text classification within text mining. This section investigates strategies and issues encountered in topic based text classification, and its relationship to data mining algorithms.

Initial text classification methods suggested in research were based on knowledge engineering approaches, where expert knowledge was elicited and encoded into sets of rules to classify documents. One example can be seen in the *Construe* system (Hayes et al, 1990) for classification of news stories. However systems based on manually built rule bases suffer from the high maintenance cost of updating the repository with expert knowledge, and slow implementation times due to manually deriving rules from experts. In the 1990s, with advances in machine learning and availability of computing

resources, this method gave way to more automatic supervised learning techniques that depend on training data, but are able to perform classification without the expensive knowledge elicitation step while achieving similar performance. In present day research, data driven machine learning methods prevail as the main paradigm for performing text categorisation (Sebastiani et al, 2001).

The most common representation of documents for text classification is based on the word vector representation, also referred to as *bag of words*, seen in the previous section. Early work on evaluating text classification with word vectors can be seen in the rule induction method proposed in (Apte et al, 1994); In (McCallum et al, 1998) the Naïve Bayes classifier is applied to text categorisation of topics in news sources; In (Joachims, 1996) a comparison of various supervised learning methods with TF-IDF word vectors is presented; A more recent example using binary presence word vectors and support vector machines applied to spam filtering is presented in (Kolcz at el, 2001). Other applications of this representation to text classification have been surveyed in the literature and can be found in (Joachims, 1998; Weiss et al, 2004; Sebastiani et al, 2001).

Extensions to the bag of words representation are also seen in the literature, aiming at capturing other useful non-textual information from documents for classification. In (Corney et al 2002), stylometric measures such as number of paragraphs, paragraph length, and types of punctuations are added to the model for gender detection of emails. In (Moschitti et al, 2004), linguistic information from parts of speech and proper nouns extracted from text are added as features to text categorisation of news sources.

As mentioned earlier, the word vector representation of documents generates data sets with very large number of attributes, which are liable to issues related to the curse of dimensionality seen in Chapter 3. One approach to mitigate this problem is to use linguistic pre-processing such as word lists, and stemming to reduce the number of terms before initiating text classification. Another approach is to employ feature selection mechanisms seen in data mining to automatically remove less relevant features while minimally affecting classification performance. Studies seen in (Rogati

et al, 2002; Forman et al, 2002) report good on using statistical feature selection methods for reducing the size of the feature space in text classification problems.

It is also worth noting that the high dimensionality of text data sets make it a good candidate for the application of support vector machines. Theoretical results seen in Chapter 3 indicate that the algorithm's training efficiency and error rate are a function of support vectors, and not dimensions, suggesting potentially better results could be obtained for this domain. In (Joachims, 1998) it is argued that in the high dimensional feature space of textual data, most features are significant and should be present in the model. The results of a feature selection experiment are shown where aggressive pruning of features led to poorer results. In the same study, superior results were obtained with support vector machines in comparison to other classification algorithms on text data sets, thus eliminating the need for the often expensive feature selection step.

## *4.2. Opinion Mining*

*Opinion Mining* is a new and exciting field of research concerned with extracting opinion related information from textual data sources. It has the potential for a number of interesting applications both in commerce and academic areas, and poses novel intellectual challenges, which continues to attract considerable research interest. In this section the research field of opinion mining is introduced, its motivations, key tasks and challenges are discussed in more details. Then, the *SentiWordNet* lexical resource for opinion mining is presented, and its potential advantages, applications and limitations are discussed.

### 4.2.1. Introduction: Opinions in Text

Information concerning people's opinions can be a very important component for more accurate decision making in a number of domains. Companies, for instance, have a keen interest in finding out what are their customers' opinions on a new product launched on a marketing campaign. Consumers on the other hand would benefit from accessing other people's opinions and reviews on a given product they are intending to purchase, as recommendations from other users tend to play a part on influencing purchasing decisions. Knowledge of other people's opinions is also important in the

political realm, where for instance, one could find out the sentiment towards a new piece of legislation, or an individual such as a politician or activist.

In recent years, the internet has enabled access to opinions in the form of written text from a variety of sources and in a much larger scale. It also made it easier for people to express their opinions on virtually any subject by means of specialised product review websites, discussion forums and blogs. This is in fact a growing trend, as pointed out in the research performed by (Horrigan, 2008) with 2000 American adults, stating that 60% of the American population, or 81% of the country's internet users have used the internet to perform research on a product they intended to purchase, as of 2007. The research also shows that over 30% of American internet users have at one time posted a comment or review online about a product or service they've purchased suggesting an ever growing availability of opinion related information on the web available to consumers. It is worth highlighting that consumer goods are not the only target of opinion related content: specialised websites that gather and provide opinion information on companies, politicians and education resources are also available. Opinion sources are not restricted to specialised review sites, and are also contained in users' blog posts, discussion forums and embedded in online social networks.

The internet is clearly a vast repository of publicly available user generated content dedicated to expressing opinions on any topic of interest. However, despite the clear benefit of having such information available, it was also pointed out in (Horrigan, 2008) that 58% of internet users reported finding product information online either confusing, difficult to find, or were overwhelmed by the volume of information available. These results indicate the problem of information overload, discussed in (Cody et al, 2002; Farhoomand et al, 2002) also exists in the realm of product reviews, where vast information resources are difficult to leverage, and are poorly utilised as a result. Automated methods for efficiently extracting knowledge from these resources appear an attractive proposition for both individuals who would be able to make better decisions and to companies who could quickly gauge opinions on their products and services, adding knowledge to their product development processes. This in fact is precisely the realm of knowledge discovery and data mining proposed in (Fayyad, et al, 1996) and discussed in Chapter 32. In addition, opinions are generally expressed in textual form, making it a rich ground for the application of text mining and techniques

to analyse natural language. Thus, the motivating need to analyse large volumes of opinion information, coupled with advances in natural language processing and machine learning methods gave rise to research in the emerging field of *Opinion Mining*.

Opinion Mining is concerned with applying computational methods for the detection and measurement of *opinion*, *sentiment* and *subjectivity* in text (Pang et al, 2008). A text document can be seen as a collection of objective and subjective statements, where objective statements refer to factual information present in text, and subjectivity relates to the expression of opinions, evaluations and speculations (Wiebe, 1990). To further illustrate the motivations for performing opinion mining, we now survey the potential applications of computing systems that apply techniques that detect and extract the subjective aspects of text.

**Search Engines**

The most direct application of opinion mining techniques would be the searching of opinions within documents. Finding out subjective statements related to a topic, and their bias can augment traditional search engines into recommendation engines by retrieving results on a given topic containing only positive or negative sentiment (Pang et al, 2002), for example when searching for products that received good reviews on a particular area, like a user query for digital cameras with good feedback on battery life. On the other hand, information retrieval systems that need to provide factual information on a given subject can detect and discard opinion information to increase the relevance of results (Wiebe et al, 2004).

**Inappropriate Content**

In a collaborative environment such as a discussion group or email list, opinion mining could be applied to classify subjective statements containing overly heated or inappropriate remarks, also called *flaming* behaviour (Kaufer, 2000). Similar techniques could assist more efficient online advertisement strategies by avoiding ad placements next to content that is related to the ad campaign, but carries unfavourable opinions towards a certain product or brand (Jin et al, 2007).

**Customer Relationship Management**

Systems that manage customer interactions can become more responsive by using sentiment detection as a tool to automatically predict the level of satisfaction of client feedback. One example is the automatic classification of customer feedback replies by email containing positive or negative sentiment (König et al, 2006), which could then be used for automatically routing of messages into the appropriate teams for corrective actions when necessary.

**Business Intelligence**

Opinion mining has the ability to add analysis of subjective components of text to discover new knowledge from data. This may take the form of aggregated sentiment bias information from user feedback which can be used to drive marketing campaigns and improve product design. In the financial industry, sentiment information present on financial news have been studied to assess its impact on the performance of securities (Devitt et al, 2007).

**Benefits to Knowledge Management**

From the examples of opinion mining applications presented above, it can be seen this field of research has the potential to add value to knowledge management efforts in companies across a range of knowledge based activities. Knowledge based systems that store explicit content can become more efficient by extending its query interface to include opinion information for more relevant results, or excluding subjective documents when more factual results are needed. Knowledge sharing systems that provision collaborative environments for exchange of explicit or tacit knowledge can become more fluid and require less administration efforts by employing sentiment detection to avoid flaming and other unwanted user behaviour; finally, knowledge discovery systems can leverage opinion information to help knowledge creation in the organisation, and improve decision making where user feedback is relevant.

### 4.2.2. Key Problems Addressed by Opinion Mining Research

In an attempt to map the activities of the emerging field of opinion mining the research survey of (Pang et al, 2008) categorises the area into two broad fields of classification and extraction. Classification would entail research related to detecting in first instance if a piece of text can be categorised as subjective or objective and, in case it is

subjective, be ability to correctly predict the text's sentiment orientation or *polarity*; the extraction aspect of opinion mining shares the concerns of information retrieval, and attempts to identify within a text document what are the key attributes of an opinion, such as the holder or to what entity it refers to, with a view to build summaries based on opinion information. A similar formulation appears in (Esuli et al, 2006), where the primary objectives of opinion mining are categorised into 1) determining the degree in which a given text is objective or subjective; 2) determining whether it expresses a positive or negative bias, if a text is indeed subjective; and 3) determining the degree of *strength* of the polarity of a given subjective text.

In (Pang et al, 2008) other categories that may fall into the realm of author opinion or sentiment are mentioned. For instance, mining "pros and cons" expressions reflect points of view part of an argument for a given topic and are close but not necessarily the same as author sentiment, and indeed a formulation of opinion proposed in (Kim et al, 2004) does take this into account. *Agreement detection* is a similar problem where differing or agreeing opinions between two distinct documents are sought.

One field of research that shares some of the concerns of opinion mining is that of *affective computing*, aiming at the development of computational approaches for detecting human emotions such as anger, fear and humour. Affective computing has applications in human computer interaction, but is closely linked to the problem of detecting subjective text, since both relate to the expression of human emotions. In (Mihalcea et al, 2005) a method for detecting humour in text is proposed with good empirical results, and in (Strapparava et al, 2004) a lexical resource for assisting the detection of emotions is proposed. Further in this chapter, it will be shown how this same resource is used as a starting point for establishing the opinion strength for terms on the *SentiWordNet* lexical resource (Esuli et al, 2006).

For the purposes of this research, the focus or this review is on the predictive aspects of opinion mining related to the tasks of subjectivity detection and sentiment classification of texts. As noted in (Pang et al, 2008), opinion extraction research can be often placed more naturally within the realm of information extraction rather than on predictive data mining. It is acknowledged however that opinion extraction is a relevant part of this field and one that often goes hand in hand with opinion detection

and classification methods, as illustrated in (Dave et al, 2003), thus any relevant research on opinion extraction will be mentioned where appropriate to the discussion.

### 4.2.3. Subjectivity Detection

In order to detect subjectivity in text automatically, a computational model requires a formalisation of what is understood by the concept. In (Wiebe et al, 2004), the subjectivity of a sentence is defined based upon previous work in linguistics and literary theory. First, there are *subjective elements*: the linguistic expressions that characterise private states of mind. Characterising subjective elements is not a trivial task; they may appear in text as single words, expressions or entire sentences, may depend on the context, and may also be evident in text style. A subjective element expresses the opinions, thoughts and speculations of a *source*, that is the document author or someone mentioned in the text. Finally, a subjective element has a *target*, or the object being referred to. A similar abstraction to subjectivity is presented on (Kim et al, 2004), where an opinion is expressed as a quadruple of the form *[Topic, Holder, Claim, Sentiment]* in which a *Holder* believes a *Claim* on a given *Topic*, with a given *Sentiment* associated with it. Subjectivity detection is generally concerned with finding the subjective elements or sentiment in text, but other aspects of the above characterisation can also be of relevance for query systems and summarisation tasks. For instance, when tracking the opinions of a given person. Handling queries such as "*what does the Taoiseach think of the new EU referendum?*" would involve knowledge of the subjective element and also the source and target components of subjectivity.

There have been several approaches proposed in the literature to detect elements of subjectivity on text. In (Wiebe et al, 2004) an approach is proposed based on exploring word relationships learned from an annotated corpus of subjective expressions. Subjectivity is annotated manually at expression, sentence and document levels, and used to train detection methods based on terms presence and term collocations, or their position in the text relative to each other. This is based on the hypothesis that subjectivity of an expression is a function of how subjective their surrounding elements in text are. The method has the advantage of relying on automatically extracting knowledge from a corpus. Word collocation analysis also assists in term

disambiguation on cases where words can alternate between objective and subjective meaning depending on context, such as the word “heart” in the examples:

- “*the gory scenes in this film are not for the faint of heart.”*
- “*open heart surgery will be available from January.”*

Analysis of the corpus also has shown that unique or rare terms are often associated with subjective expressions, indicating a certain measure of author “creativity” when expressing opinions.

Another approach to detecting subjectivity is proposed on (Pang et al, 2004), where machine learning classification algorithms are trained to predict objective or subjective sentences based on a training set of extracted documents from the internet. The subjective data set is comprised of 5000 text extracts from film reviews, whereas the objective set is built from 5000 extracts from film plot summaries. A similar approach is presented in (Yu et al, 2003) where a Naïve Bayes classifier is trained to detect subjective documents and based on a data set of news sources known a priori to carry objective (news and business sections) and subjective (editorials and letters to the editor) content, with good results. The method is extended to sentence-level opinion detection by including parts of speech, sentence similarity measures and counting the presence of semantically oriented terms from a subset of manually labelled seed words. Results from (Wiebe et al, 1999) also show positive results on Naïve Bayes classifiers trained a data set of subjective and objective documents, using features derived from part of speech, punctuation and syntax elements.

### 4.2.4. Sentiment Classification

Sentiment classification is concerned with determining what, if any, is the sentiment *orientation* of the opinions contained within a given document. It is assumed in general that the document being inspected is known to represent opinion, such as a product review, and that the document’s opinion refers to a single entity (Pang et al, 2008). Opinion orientation can be classified as belonging to opposing positive or negative polarities – positive or negative feedback about a product, favourable or unfavourable opinions on a topic – or ranked according to a spectrum of possible opinions, as is the

case with film reviews with feedback ranging from zero to five stars (Pang et al, 2005). Sentiment classification is at the centre of this dissertation's experiment, and in this section, the approaches to sentiment classification studied in the literature and its challenges are surveyed in more details.

**Word Vectors**

One natural approach to performing sentiment classification is to take the traditional text mining representation of documents as word vectors, where each entry maps to a term found in the corpus of documents, and the value of a given entry corresponds to a measure of term presence or a measure of relative term frequency from the field of text mining and information retrieval. In (Pang et al, 2002) a series of experiments using various classes of word vectors for sentiment classification of film reviews generated positive results for single term word vectors – or *unigrams* - using binary presence values for each term. Binary presence did perform better than frequency-based word vectors, suggesting that term existence, rather than frequency is more significant to opinion identification. This distinction is observed in (Pang et al, 2008), with a suggestion that traditional topic-based document classification relies more strongly on repeated occurrences of the same terms throughout the text, whereas this may not be the case for opinions.

The experiment in (Pang et al, 2002) achieves best results when using term unigrams rather than larger n-gram features, even though bigrams could capture sentiment encoded in for form of 2-term expressions such as "really good" or "much preferred", etc. The poorer classification results could be attributed to a necessary increase in the volume of training data for all relevant term n-grams to be captured. Indeed, work from (Cui et al, 2006) reports good results for higher order n-grams where a significantly larger training data set comprised of over 320.000 product reviews is available. Another similar experiment based on word vectors and product reviews as the data set reports good results for tri-grams is seen in (Dave et al, 2003).

Taking the traditional text mining approach to train a classifier based on word vectors for opinion mining generates good classification performance results, but as observed in (Pang et al, 2008), these results stay well below those obtained for topic-based document classification using the same techniques. Empirical performance metrics for

topic-based text categorisation using Support Vector Machines seen in (Joachims, 1997) and surveyed in (Sebastiani, 2002) show how high precision, high recall topic based classification can be achieved, based on results using well known experiment data sets. This observation, coupled with further analysis of opinion bearing documents suggests that sentiment information needs to be captured by other means. One point highlighted on (Pang et al, 2002) is the issue of *thwarted expectations*, as seen on the extract below:

> *"This film should be brilliant. It sounds like a great plot, the actors are first grade… However, it can't hold up"*

In the above case a sentence contains a high number of positive statements, building up the expectation of a positive review, but the overall sentiment of the review is still negative. This affects prediction decisions based on term information presence alone, and suggests that the order of which opinions are presented is of importance to overall sentiment (Pang et al, 2008).

**Word Sense Disambiguation**

Issues stemming from ambiguity in word sense surveyed in Section 4.1.2 of this chapter also arise on opinion mining problems. In (Wiebe et al, 2006), subjectivity detection is improved by adding a subjective feature to detect terms in need of disambiguation, and the authors speculate improvements to sentiment classification tasks with the assistance of term disambiguation techniques. This need has also been highlighted in (Dave et al, 2003), when inspecting results of sentiment classification experiment based on term opinion information.

**Parts of Speech**

Classifying terms from a textual document into its grammatical roles, or parts of speech within a sentence has also been explored in opinion mining. A motivating factor behind this approach is that detecting parts of speech can be considered a form of word disambiguation for the cases where word senses are associated with its grammatical use, such as noun, verb, etc (Wiebe et al, 1998). Another factor is the finding that adjectives are considered good indicators of opinion information and have been seen to provide good correlation to sentiment orientation, as reported by (Turney,

2002). In (Pang et al, 2002), a study reports good results using only adjective words as features to perform sentiment classification using a machine learning method, however with poorer results than using full word vectors as features. The use of parts of speech as a pre-processing step for deriving features for opinion mining has also been seen in a number of other sentiment classification experiments: In (Yu et al, 2003) it is used as part of a feature set for performing sentiment classification on a data set of newswire articles, with similar approaches attempted in (Pang et al, 2002; Salvetti et al, 2004; Gamon, 2004) on various data sets; On (Turney, 2002) a method that detects and scores patterns in part of speech is applied to derive features for sentiment classification, with a similar idea applied to opinion extraction for product features seen in (Yi et al, 2003).

**Document Style and Document Structure**

The subjective components of a document also have been shown to have relationships to the document structure and writing style, as in the example of *thwarted expectations*, previously discussed in this section. One consideration is *term position* within the document. It can be argued that the location of a specific opinion bearing term within a document can have greater or lesser influence in overall sentiment classification: if for instance, this term is placed towards the end of the document, it may have a greater relation to author's opinion, as the end of the document is generally where concluding remarks are present. This is one aspect that has been explored in the experiments presented in (Pang et al, 2002), where it was seen to influence overall classification, albeit in a small scale. Another attempt to model discourse structure can be seen in (Devitt et al, 2007) where a graph-based representation of text relationships is proposed, based on linguistic models of lexical cohesion and other metrics extracted from the document.

Detecting the existence of expressions that can increase, decrease or invert sentiment orientation of text - also called *valence shifters* - are of importance to sentiment classification of documents. A comment present on a film review, such as the one below:

> *"This film is not great, not funny and not interesting."*

Predicting the correct sentiment of the above review can not rely on term orientation alone, since each positive-oriented term has been negated and is expressing its exact opposite. In (Pang et al, 2002) negation detection is modelled by adding a modifier prefix to negated terms, such as converting “great” into “NOT_great”. The resulting modified text is then used as input for a word vector classifier. Several approaches have been studied for the detection of negation in the context of extracting information from medical records (Chapman et al, 2001; Huang et al, 2007; Mutalik et al, 2001). Other valence shifting modifiers, such as “very”, “just” or “extremely” have also been shown to influence sentiment classification of the overall document (Kennedy et al, 2006).

Humoristic features such as sarcasm and irony also do play a part in expressing author sentiment. These can be relatively more complex to identify, usually not depending on term sentiment alone, but relying on word play, contrasts and domain knowledge. Other affective expressions such as anger, joy, and fear can also be closely related to author sentiment, and therefore opinion. The role of affective computing to sentiment analysis has been highlighted in (Strapparava et al, 2006). Supervised approaches to humour detection have been investigated in (Mihalcea et al, 2005) for a limited aspect of written humour, but with some success when experimented on a test data set.

Finally, it is to be expected that opinionated documents may contain also objective sections. On product reviews this may amount to sections describing the product features, as opposed to expressing an opinion on them; on film reviews the author may chose to present details of the plot, or the background of a certain actor, to further back up an argument. It can be speculated that the objective sections of an opinionated document in general will carry less opinion bias than the subjective ones, and may cause a decrease in performance on overall document classification. As an example we can consider for instance the case where an actor dialogue is inserted by the author into a film review, containing terms in opposition to author opinion. Similar issues have been noted in (Pang et al, 2002; Kennedy et al, 2006), and the beneficial aspect of subjectivity detection and filtering to sentiment classification has been noted in (Pang et al, 2008). In (Pang et al, 2004) an approach to filter out objective sentences as a pre-processing step to document classification is proposed based on training on a data set

of subjective sentences, with considerable improvements over a baseline machine learning classifier.

**Combining Approaches**

Taking the view that different methods for performing sentiment classification capture different types of sentiment related information from a document, it is worth noting the contribution in the literature to combining results from more than one classifier in order to obtain better results. This can be done not only to address induction bias from a specific classifier algorithm, as seen on section 3.3.7, but also to make better decisions from a pool of classification techniques, each leveraging different types of data. Applications of this idea to sentiment classification can be seen on (Kennedy et al, 2006), where a combination of classifiers using word vectors and scores from a word list generate improved results over a baseline. A similar approach is seen on (Mullen et al, 2004) with the combination of proximity metrics and term relationships extracted from a lexicon.

In the next section we turn our attention towards a distinct class of techniques for performing sentiment classification, based on building lexicons containing sentiment information that can be employed to the detection of opinion in documents.

### 4.2.5. Lexical Resources for Opinion Mining and SentiWordNet

One common approach in performing both subjectivity detection and sentiment classification involved the use of key words that are assumed to be indicative of either positive or negative bias, and therefore also of overall subjectivity. This idea is based on the hypothesis that words can be considered as a unit of opinion information, and several methods based on this assumption have been proposed with considerable success: (Turney et al, 2003) proposes a subjectivity detection method that extends a list of seed words based on a proximity measure to other common terms in text; In (Pang et al, 2002) an experiment with a list of manually created positive and negative words yields accuracies of 69% in the task of sentiment classification of film reviews. In (Kennedy et al, 2006), a lexicon of positive, negative and valence shifter terms is built from various sources to perform document-level sentiment classification.

One interesting aspect of approaches based on word lists is that it does not necessarily require training data for making predictions, since it relies only on a pre-defined sentiment lexicon, thus being applicable to cases where no training data is present. For this reason these methods are often labelled as *unsupervised learning* approaches (Pang et al, 2008).

Creating word lists manually however is time consuming, and approaches have been proposed in the literature for automatically creating resources that contain opinion information on words based on readily available lexicons, often termed *lexical induction* (Pang et al, 2008). In (Kennedy et al, 2006) a lexicon of positive, negative and valence shifting terms is built from various data sources for the purposes of sentiment classification. Another common approach is to derive opinion information from the freely available WordNet database of terms and relationships (Miller et al, 1990), typically by examining term relationships to a subset of core terms assumed a priori to carry opinion information, such as “good”, “excellent”, “bad” and “poor”. This approach to lexical induction can be seen on subjectivity detection research conducted in (Yu et al, 2003) and in sentiment classification (Dave et al, 2003; Kim et al, 2003; Salvetti et al, 2004). A similar example of WordNet-based lexicon has been proposed for the purposes of affective computing, such as the WordNet-Affect resource (Strapparava et al, 2004).

**SentiWordNet**

One example of a lexical resource conceived to assist in opinion mining tasks is *SentiWordNet* (Esuli et al, 2006). SentiWordNet aims at providing term level information on opinion polarity by deriving this information from the WordNet database of English terms and relations (Miller et al, 1990) in a semi-automatic fashion.

For each term in WordNet, a positive and a negative score ranging from 0 to 1 is present in SentiWordNet, indicating its polarity, with higher scores indicating terms that carry heavy opinion bias information, whereas lower scores indicate a term being less subjective. The table below illustrates a score for the term “interesting” extracted from SentiWordNet’s web interface.

| | |
|---|---|
|  | Term: ***Interesting***<br>o Positive: 0.325<br>o Negative: 0.0 |

**Figure 11 - SentiWordNet Sample Score (http://sentiwordnet.isti.cnr.it)**

**WordNet**

WordNet is a lexical database for the English language where terms are organised according to their semantic relations (Miller et al, 1990). It has been widely applied to problems in natural language processing with a comprehensive list of work in the literature available on (Csomai et al, 2008). Several opinion classification methods in the literature are based upon it (Kim et al 2004; Dave et al, 2003; Salvetti et al, 2004). Before describing how SentiWordNet is built, a brief discussion on the database that originated it will be of help in understanding the underlying motivations and how the data is organised.

The WordNet lexicon is the result of research efforts in linguistics and psychology at Princeton University on better understanding the nature of semantic relations of terms in the English language, and on providing a complete lexicon in the English language where terms can be retrieved and explored according to concepts and their semantic relationships. At its third version, WordNet is available as a database, searchable via web interface or via a variety of software APIs, providing a comprehensive database of over 150.000 unique terms organised into more than 117,000 different meanings (WORDNET, 2006). WordNet also grew with extensions of its structure applied to a number of other languages (WORDNET, 2009).

**Key Term Relationships**

The key relation between terms in WordNet is similarity of meaning, or *synonymy*. Terms are grouped together into sets of synonyms called *synsets*. The general criteria for grouping terms together into a synset is whether a term used within a sentence on a specific context can be replaced by another term on the same synset without modifying the sentence's understanding. One direct implication of this structure is that terms must also be differentiated by syntactic categories, since nouns, adjectives verbs and

adverbs are not interchangeable within a sentence. Synsets also contain a short descriptive text defining its terms – or gloss – to assist in specifying its meaning. This is particularly useful on synsets with only a single term, or synsets with a small number of relations.

Another important term relationship present in WordNet is *antonimity*, or whether terms are conceptually opposites. In the special case of adjectives, there is a distinction between direct and indirect antonyms, or when terms can be categorised as direct opposites, or indirectly via another conceptual relationship (Fellbaum et al, 1990). The words "wet/dry" are qualified as direct antonyms, however "heavy/weightless" are conceptually opposites and thus indirect antonyms, since they belong to synsets where a direct antonym exists between the terms ("heavy/light") but are not directly correlated.

*Hyponymy* is another class of relationship present in WordNet, and indicates a hierarchical "is-a" type of relationship between terms, such is the case with "oak/plant" and "car/vehicle", while *meronymy* relationships indicate "part-of" types of relationship between terms. For the special case of adjectives, an *attribute* type of relationship exists, indicating of what generic attribute the adjective is a modifier, for example the example adjectives "heavy" and "light" are modifiers of the attribute "weight". WordNet would then link the noun representing the attribute to the adjectives that modify it with this type of relationship.

**Building SentiWordNet**

Building on the strengths of WordNet's semantic relationships, SentiWordNet derives opinion scores for synsets using a semi-supervised method where only a small portion of synset terms - called the *paradigmatic* terms - are manually labelled, with the remaining database derived using an automated method. The complete process is described in (Esuli et al, 2006) and summarised below:

1. Manually label paradigmatic terms extracted from the WordNet-Affect lexical resource (Strapparava et al, 2004) into positive or negative labels, according to opinion polarity.

2. Iteratively expand each label by adding terms from WordNet that are connected to already labelled terms by a relationship considered to reliably preserve term orientation. The following relationships are used to extend the labels:
    a. Direct antonym
    b. Attribute
    c. Hyponymy (pertains-to and derive-from)
    d. Also-see
    e. Similarity

3. From newly added terms, add to opposite label the terms containing directly opposite opinion orientation, according to the direct antonym relationship.

4. Repeat steps 2 and 3 for a fixed number of iterations *K*.

Upon completion of steps 1-4, a subset of WordNet synsets is now labelled either positive or negative. To complete the score assignment for all terms, a set of classifiers is trained on their synset *glosses*, or textual definitions of each synset meaning available on WordNet. The process continues by classifying new entries according to this training data, and generating an aggregated score, as detailed below:

5. For each labelled synset from steps 1-4, produce a word vector representation, along with a positive/negative label. This data set is used to train a committee of classifiers built as follows:
    a. Train a pair of classifiers to make the following predictions: *positive/non-positive*, and *negative/non-negative*.
        i. synsets that belong to both positive and negative labels are excluded from the training set and assigned to the “objective” class, with zero-valued positive and negative scores.
    b. Repeat process for different sizes of training sets. These are obtained by varying K in the previous stage: 0,2,4 and 6.
    c. For each training set, use Rocchio and Support Vector Machine classification algorithms.

6. When applying the set of classifiers to new terms, each resulting classifier returns a prediction score as a result. These summed together and normalised to 1.0 to produce the final positive and negative scores for a term.

The process for building SentiWordNet illustrated above highlights the reliance of term scores on two distinct factors: the choice of paradigmatic words that will generate the initial set of positive and negative scores must be carefully considered, since the extension of scores to the remainder of WordNet terms relies on this core set of terms for making a scoring decision. Secondly, the process relies on synset's textual description, or glosses, for the machine learning stage of the process, to derive a new term's similarity to positive or negative terms.

**Applying SentiWordNet**

Earlier in this section the advantages of lexicon-based approaches to opinion mining were observed, and results from experiments on both subjectivity detection and sentiment classification were investigated. The use of SentiWordNet as a lexical resource for opinion mining could be of advantage on various instances. The approach of using individual terms as a unit for sentiment information has received considerable research attention in opinion mining, and SentiWordNet could be applied as a replacement to manually building sentiment lexicons from WordNet, often done on an ad-hoc basis for specific opinion mining research, as found on (Salvetti et al, 2004; Dave et al, 2003; Kim et al, 2004). Validating automated methods for building term orientation information such as SentiWordNet can be useful in the scalability and automation of these approaches to opinion mining.

## *4.3. Conclusion*

In this chapter the research areas of text mining and opinion mining were surveyed. Text mining concerns the computational treatment of text for extraction of novel information, and leverages techniques from machine learning, natural language processing, information retrieval and computational linguistics. Applications of text mining to knowledge discovery were surveyed, based on exploratory analysis and other traditional data mining techniques.

The representation of documents for performing text mining was studied in more details, with the word vector, or bag of words method being one of the most popular methods for representing text for machine learning, demonstrating very effective empirical results.

The research area of opinion mining was introduced. Opinion mining is a new field of research leveraging components from data mining, text mining and natural language processing, and a wide range of applications of extracting opinion from documents is possible, as discussed in this chapter. These range from improving business intelligence in organisations to information retrieval systems, recommender systems and more efficient online advertising and spam detection. It was seen that opinion mining can be beneficial to knowledge management initiatives either directly, by improving the quality of knowledge repositories through opinion-aware features, or by adding to the knowledge that can be extracted from textual data sources, thus indirectly creating more opportunities for knowledge creation within the company.

Finally, the WordNet and SentiWordNet lexical resources were introduced, with a presentation of its building blocks and potential uses. SentiWordNet is an extension of the popular WordNet database of terms and relationships, and is a readily available lexical resource of term sentiment information, which could be used on opinion mining research where a number of similar approaches were devised in an ad-hoc fashion. SentiWordNet is also one key component of this dissertation's research. In the next chapter, the capabilities and structure of this resource are explored in details having in mind the challenges to opinion mining surveyed in this chapter. The end result is the design of a set of features that leverage sentiment information extracted from SentiWordNet and can be applied to sentiment classification problems.

# 5. DESIGNING FEATURES WITH SENTIWORDNET

## *5.1. Introduction*

As outlined in the introductory discussion on the research problem and objectives in Chapter 1, this dissertation's experiment is comprised of two distinct parts: First, in order to use SentiWordNet as a tool for performing sentiment classification, a set of features that capture as much sentiment information as possible from textual documents needs to be devised. Then, once a feature set is generated from text documents with SentiWordNet, these can be used as input to a classifier algorithm and results on classification performance and execution speed can be analysed. This chapter begins the experiment discussion by presenting the first aspect of this experiment.

In this chapter the structure of the SentiWordNet database is analysed in detail and the Polarity data set of film reviews is presented. Considerations on writing style are presented, and the implications to the type of information SentiWordNet can extract for the purposes of opinion mining are discussed. These considerations will drive the requirements for data preparation and cleanup of the source text needed to generate an effective data mining exercise, and as noted in (Shearer, 2000) and surveyed in Chapter 3, these are important contributing factors for the success of knowledge discovery activities.

The outcome of this chapter is a specification for a set of features that takes a film review in plain text as the starting point and captures sentiment information present on terms using SentiWordNet. This set of features can be used as input to train supervised learning methods in performing sentiment classification.

## *5.2. The SentiWordNet Database*

As detailed on Section 4.2.5 of the previous chapter, SentiWordNet is a database containing opinion scores for terms derived from the WordNet database version 2.0. It is built using a semi-supervised method to obtain opinion polarity scores from a subset

of seed terms that are known to carry opinion polarity. Each set of terms sharing the same meaning, or *synsets*, is associated with three numerical scores ranging from 0 to 1, each indicating the synset's objectiveness, positive and negative bias. One important characteristic of SentiWordNet is that positive and negative scoring is *graded* for any given term, and it is possible for a term to have non-zero values for both positive and negative scores, according to the following rule:

For a synset *s*, we define:

- *Pos(s)* → Positive score for synset *s*.
- *Neg(s)* → Negative score for synset *s*.
- *Obj(s)* → Objectiveness score for synset *s*.

Then the following scoring rule applies:

$$Pos(s) + Neg(s) + Obj(s) = 1$$

The positive and negative scores are always given, and objectiveness can be implied by the relation:

$$Obj(s) = 1 - (Pos(s) + Neg(s))$$

### 5.2.1. Database Structure

The SentiWordNet database is provided as a text file where term scores are grouped by synset and the relevant part of speech. The table below describes the columns for one entry in the database reflecting opinion information of a synset.

| **Field** | **Description** |
|---|---|
| **POS** | Part of speech associated with synset. This can take four possible values:<br>• a = adjective<br>• n = noun<br>• v = verb<br>• r = adverb |
| **Offset** | Numerical ID which associated with part of speech uniquely identifies a synset in the database. |

| | |
|---|---|
| **PosScore** | Positive score for this synset. This is a numerical value ranging from 0 to 1. |
| **NegScore** | Negative score for this synset. This is a numerical value ranging from 0 to 1. |
| **SynsetTerms** | List of all terms included in this synset. |

**Table 9- SentiWordNet Database Record Structure**

To illustrate how opinion information appears in SentiWordNet, the table below presents sample rows extracted from the raw database file.

| POS | Offset | PosScore | NegScore | SynsetTerms |
|---|---|---|---|---|
| a | 1001456 | 0.375 | 0.125 | casual<br>everyday |
| n | 13488485 | 0.0 | 0.125 | pull<br>twist<br>wrench |
| v | 1248670 | 0.125 | 0.0 | truss<br>tie_up<br>bind<br>tie_down |
| r | 326136 | 0.375 | 0.25 | dreamily<br>dreamfully<br>moonily |

**Table 10 - Sample SentiWordNet Data**

### 5.2.2. Statistics on Part of Speech Scoring

As seen on the previous section, SentiWordNet terms are categorised by the role being played in a given sentence, or the part of speech the term is used as. To further understand how opinion scores are affected by part of speech, the table below reproduces analysis presented on (Esuli et al, 2006):

| Part of Speech | % Synsets with Objectiveness = 1 | Average Objective Score | Average Pos. Score | Average Neg. Score |
|---|---|---|---|---|
| Noun | 83.50 % | 0.944 | 0.022 | 0.034 |
| Verb | 81.05 % | 0.940 | 0.026 | 0.034 |
| Adverb | 32.97% | 0.698 | 0.235 | 0.067 |
| Adjective | 44.71% | 0.743 | 0.106 | 0.151 |

**Table 11 - Scoring statistics per part of speech (Esuli et al, 2006)**

It can be seen from the above data that nouns and verbs are predominantly objective in nature, and carry little positive or negative bias. According to the process of building SentiWordNet, this indicating nouns and verbs have weaker relations to other WordNet terms known to have either positive or negative bias, while adverbs and adjectives are the parts of speech carrying the highest percentage of terms with a non-negative subjective score. As observed on (Esuli et al, 2006), it is an indication that the use of *modifiers* (adjectives or adverbs) is more frequent when expressing subjective opinion than speech parts such as verbs and nouns, more commonly used to denote entities of objective nature. Another important observation is that while adverbs do carry considerable polarity weight (only 32.97% of terms contain no subjective bias), the average scoring tends to be overwhelmingly positive.

## *5.3. Considerations on SentiWordNet Data*

After analysing the database structure of SentiWordNet, this section explores key aspects that need to be taken into consideration when designing features to be used in sentiment classification.

### 5.3.1. Automatic Part of Speech Tagging

Data in SentiWordNet is categorized according to part of speech, and indeed as seen on Table 11, there are considerable differences in the level of objectiveness a synset might carry, depending on its grammatical role. Information on part of speech in the source documents being classified will need to be extracted, so that SentiWordNet scores can be accurately applied. To achieve this, a *part-of-speech tagging* algorithm can be employed to automatically classify words into categories based on parts of speech from the source documents. Part-of-speech taggers and their use within opinion

mining research were discussed on section 4.2.4 of the review on opinion mining research literature.

A part-of-speech tagger receives as input a plain text document, and returns as output a document where every word and punctuation mark is associated with a tag that indicates the part of speech the term is used as. For example, the input sentence:

> "the variety of music , as well as the beautifully shot performances , are easy to become immersed in ."

Generates the following output from a part-of-speech tagger:

> " the/DT variety/NN of/IN music/NN ,/, as/IN well/RB as/IN the/DT beautifully/RB shot/VBN performances/NNS ,/, are/VBP easy/JJ to/TO become/VB immersed/VBN in/IN ./. "

Each term has been associated with a relevant tag indicating its role in the sentence, such as verb, noun, adjective, etc. Several standards exist for tag formats, of which the most popular are related to the *Penn Treebank* annotated corpus (Marcus et al, 1993) and the various instances of the CLAWS tag sets, derived from the original tag set for the brown corpus (Garside, 1987). To illustrate the above example, the table below highlights key tags from the Penn Treebank tag set relevant to SentiWordNet, with the complete set of tags available in the appendix section.

| **Part of Speech** | **Penn Treebank Tags** |
|---|---|
| Adjective | JJ, JJR (Comparative), JJS (Superlative) |
| Verb | VB, VBD (Past tense), VBP (Present tense), VBZ (Present tense 3rd person), VBG (Gerund), VBN (Past participle). |
| Adverb | RB, RBR (Comparative), RBS (Superlative) |
| Noun | NN, NNP (Proper noun), NNPS (Proper noun, plural), NNS (Plural) |

**Table 12 - Penn Treebank Tags (Marcus et al, 1993) for parts of speech present in SentiWordNet**

After tagging plain text documents, this information needs to be parsed so that it can be used with the SentiWordNet database. This process will require the development of an application that reads a tagged document and correctly match terms and their part of speech tag to a SentiWordNet score.

### 5.3.2. Word Sense Disambiguation

When evaluating scores for a given term using SentiWordNet, an issue arises in determining to what specific WordNet synset the term belongs to and which score to take into account. Consider the example for the term “mad”, with four synsets in WordNet.

| **Synset** | **SentiWordNet Score (Pos, Neg)** | **Gloss** |
|---|---|---|
| huffy, mad, sore (roused to anger) | (0.0, 0.125) | "she gets mad when you wake her up so early"; "mad at his friend"; "sore over a remark" |
| brainsick, crazy, demented, disturbed, mad, sick, unbalanced, unhinged (affected with madness or insanity) | (0.0, 0.5) | "a man who had gone mad" |
| delirious, excited, frantic, mad, unrestrained (marked by uncontrolled excitement or emotion) | (0.375, 0.125) | "a crowd of delirious baseball fans"; "something frantic in their gaiety"; "a mad whirl of pleasure" |
| harebrained, insane, mad (very foolish) | (0.0, 0.25) | "harebrained ideas"; "took insane risks behind the wheel"; "a completely mad scheme to build a bridge between two mountains" |

**Table 13 - Example of multiple scores for the same term in SentiWordNet**

In the above example, there are four possible choices of meaning for the adjective “mad”, one of which refers to positive states of emotion, and carries a positive score in SentiWordNet, raising the question of which one to apply when scoring this term inside a given document. Determining which synset needs to be applied on a specific context is analogous to the problem of word sense disambiguation. In section 4.2.4 of the opinion mining literature review, this area of research and its connection to opinion mining was explored in more details. For the purposes of this dissertation’s

experiment, no sophisticated techniques of word sense disambiguation are being considered due to time constraints and the impact on the complexity of the data set being modelled. At first instance, some level of disambiguation can be obtained from part of speech information as noted by (Wilks et al, 1998), and part of speech tagging is already executed as a requirement for extracting SentiWordNet scores. If however the process is faced with the task of scoring a term with multiple senses and same part of speech, a simpler approach will be taken, based on the following rules:

- Evaluate scores for each synset for a given term;

- If there are conflicting scores – e.g. positive and negative scores exist for the same term – calculate the average of all positive scores and all negative scores, and

- Return the averaged SentiWordNet score with higher value only if the positive and negative scores differ by more than a given threshold.

This approach assumes that ambiguous synsets with a majority score in a given orientation are likely to appear more frequently in the document and are therefore chosen. If the aggregated positive and negative scores for an ambiguous synset are below a given threshold, it is assumed then that a decision can not be made on term orientation and the score is discarded. Because word sense is not evaluated in depth, this approach may limit the amount of information gathered from SentiWordNet at the expense of some discarded scores, and as seen on the above example, may not always guarantee the correct scoring is being applied. It is hoped these will not be significant to the overall performance of the method, however future developments of the SentiWordNet model taking into account more sophisticated techniques of word sense disambiguation could yield positive results.

### *5.4. The Polarity Data Set*

The polarity data set is a set of film review documents available for research in sentiment analysis and opinion mining. It was first introduced as a research data set along with Bo Pang and Lillian Lee's initial results on machine learning methods for

sentiment classification presented in (Pang et al, 2002). The most recent available data set is version 2.0, and is the one being used for this dissertation's experiment. It comprises 1000 positive labelled and 1000 negative labelled film reviews extracted from the Internet Movie Database Archive (Pang et al, 2004). In this section, the polarity data set is further evaluated with considerations on how SentiWordNet can be used to extract opinion bias information from documents contained in it.

### 5.4.1. Document Structure

A film review from the polarity data set already underwent several pre-processing tasks aiming at standardising the text (Pang et al, 2004):

- All text is converted to lowercase.
- Each line in a document corresponds to a single sentence.
- All HTML tags are stripped from the document – e.g. documents are plain text.
- Ratings information is removed from text: Labels are derived from rating information explicitly mentioned in the document. This information is removed from the data set since author bias should be indirectly implied from the text, and not from the rating scale given.

Labelling a document as positive or negative is derived from ratings information explicitly stated in the review. Because the formatting of ratings in text is inconsistent, a set of ad-hoc rules was derived for deciding on the correct label. The rules are documented in the *readme* file for version 2.0 of the data set, and consist of:

- Ratings must be explicitly mentioned as a numerical or "star" scale. Valid examples are: 8/10, nine out of ten, three stars (out of five), etc.
- In a five-star system, positive labels are assigned to ratings of "three-and-half stars" or higher, whereas negative labels are assigned to ratings of "two stars" or lower.
- In a four-star system, positive labels are assigned to ratings of "three stars" or higher, whereas negative labels are assigned to ratings of "one-and-half stars" or lower.

- In a letter grade system, “B” or above is considered a positive review, whereas “C-“ or lower is considered a negative review.

**Document Statistics**

The table below presents document statistics to assist in understanding the document structure for a typical review, for each document class. *Average terms/Doc* counts all of the terms in a document and averages the results for all documents in a class; *Average sentences/Doc* calculates a similar metric for sentences; *Unique Terms/Doc* counts each term in a document only once. Finally, *term-to-sentence ratio* averages the number of terms in a sentence for a given document, then averages the result for all documents in each class. Further considerations on the below statistics will be made while applying SentiWordNet features and evaluating results obtained.

| **Class** | **Average Terms/Doc** | **Average Sentences/Doc** | **Average Unique Terms/Doc** | **Average Term-to-Sentence Ratio** |
|---|---|---|---|---|
| Positive | 685.526 | 35.941 | 351.973 | 19.208 |
| Negative | 611.903 | 35.419 | 326.641 | 17.968 |

**Table 14- Polarity data set document statistics**

### 5.4.2. Considerations on Writing Style

To extract scores from a review using SentiWordNet, a document needs to be scanned, and each term would receive a score based on SentiWordNet data and part of speech information. Further investigation on writing style reveals other parameters are also relevant in determining how to score terms appropriately. Based on challenges to opinion mining previously discussed in section 4.2.4, this section explores these considerations and how to address the potential issues affecting sentiment classification.

**Negation**

The use of negating terms such as “not” and “no” play a part in determining the orientation of a term. Consider the following simple examples:

- “This film is good.”
- “This film is not good.”

Clearly, both contain the term “good”, which carries positive connotation and a positive SentiWordNet score. The second sentence however has a negative meaning. Therefore a scoring method that simply adds scores for terms as they appear on text can lead to poor results.

In the English language, negation can occur in a variety of often subtle ways. In general, it involves a negation signal, a set of negated concepts for which the signal has scope on, and in some cases a supporting pattern or expression commonly appears with a type of negation (Huang et al, 2007). On the above examples, “not” indicates the negation signal. It is also worth noting that negation can modify strength of a sentence, and modify the scope of previous concepts in a sentence, such as in the examples:

- “The film is not one bit good.”
- “Production quality and good acting were absent in this film.”

In the first case, the pattern “one bit” increases the strength of the statement (e.g. “the film is not at all good”), while on the second case, the negation signal “absent” modified the concepts preceding it.

Apart from performing rich linguistic analysis on text, it is difficult to predict and correctly determine all negation cases in a text (Mutalik et al, 2001). However, effective approaches have been suggested based on simpler techniques that use text parsing rules coded as finite state regular expressions, as seen on the *NegEx* algorithm (Chapman et al, 2001) and the *NegFinder* algorithm (Mutalink et al, 2001), while other hybrid approaches that also include detecting patterns from the parsing tree of a sentence were proposed in (Huang et al, 2007). These approaches have provided good results in predicting common negation cases on objective document corpora from the medical industry.

A similar approach for dealing with negation statements in film reviews could be applied to the Polarity data set. An algorithm that detects negation by regular expressions, based on the *NegEx* algorithm (Chapman et al, 2001) is proposed, with source Python code available in the Appendix B.1 section. NegEx works by identifying three classes of expressions: There are pseudo-negating terms, where a negation expression is found but does not alter the orientation of terms; and expressions that negate previous or next terms in a sentence. If a negating expression is found, then sentiment polarity of a sentence is inverted for all terms within a specific window, or until a punctuation or negation altering term is found. A negating window is a numeric parameter that indicates the scope of a negating term within a sentence.

An interesting by-product of a negating algorithm is the ability to determine for a given text how many terms are being negated, and how often negating expressions are used as a narrative device. This information could be used as features in detecting sentiment orientation on the basis of writing style, and can be of assistance on the sentiment classification task.

**Objective Sentences**

Another aspect of a typical review is the presence of both opinion related comments and sections containing more objective text, such as a plot description, remarks on an actor or director's career. An example can be seen on the sample extract below, from one of the reviews of the Polarity data set:

> *"when the aliens begin to menace society , it may take the scientists' combined efforts to stop them before they terminate the evolution of another life form : humanity ."*

In this case, even though there are terms that may carry opinion information such as "menace" and "terminate", the sentence is very objective and does not provide evidence of author opinion. The problem can become more complex for adequate scoring since author opinion can appear in the middle of a descriptive sentence. For the purposes of this experiment, no formal detection of subjectivity in sentences is performed. To mitigate the effect of potentially objective sentences an approach based on determining areas of document more likely to be subjective is proposed below.

**Document Segmentation**

As observed in section 4.2.4, the strength of sentence opinions can be related to its position in a document, suggesting a relationship between document structure and author sentiment. In order to better detect those sentiment *hot* areas within the document, a text document can be separated into individual segments, and term scores calculated separately for each of these segments. This approach can also help in detecting areas of the document which tend to carry generally objective content, and thus of little relevance to sentiment classification. In addition, to further test the idea of term importance as a function of document position, scores can be adjusted according to a function of term position, indicating which areas of the document are more relevant.

## *5.5. Proposed Model*

In the previous sections of this chapter, the structure of the SentiWordNet database, and the polarity data set were assessed in details, and considerations were made on challenges and limitations of what opinion information can be gathered. With those in mind, a model can be proposed for creating a set of features for opinion classification using SentiWordNet. It is worth highlighting here that, as noted in (Kennedy et al, 2006; Pang et al, 2008), lexical resources such as SentiWordNet are built independently of the data set being analysed, and could be used in an unsupervised fashion, thus discarding the need for training data. The approach for a feature set proposed in this section however starts from the principle that the features obtained through SentiWordNet capture diverse aspects of document sentiment, and are best suited for the creation of a data set that can be applied to train a classifier algorithm, like other machine learning methods proposed in opinion mining. The high level approach for obtaining these features is presented in the diagram below.

distinguish document areas more likely to represent overall opinion content. Finally, a negation detection algorithm should be in place to enable the inversion of scores for a term when appropriate. The feature design is divided by feature type, and their concept is explained below.

**Overall Scoring per Part of Speech**

Intuitively, the overall positive and negative scores for all terms in a document extracted from SentiWordNet terms can be taken as a measure of opinion polarity. A similar approach is seen on (Kennedy et al, 2006). In addition, for each part of speech, the overall sum of SentiWordNet scores in a given document can also be calculated. The scoring of terms is calculated according to a function of term position within the document, as described in section 5.4.2.

**Score *Strength* Measures per Part of Speech**

Overall scoring alone may be assisted by a measure of opinion strength, which captures how strong, on average are the positive and negative scores found in the document, for each part of speech. The calculation is done by computing the total positive/negative scores divided by total positive/negative terms found, for each part of speech.

**Positive and Negative Ratios**

For each part of speech, this metric calculates the percentage of positive and negative occurrences out of total terms found, to give an indication of positive and negative term usage within the document.

**Scores per Document Segment.**

To evaluate the contribution of individual document areas to overall sentiment, each document is divided into $N$ segments of equal size, and for each segment the total positive and negative scores for a given document segment, per part of speech. For the case of adjectives, other metrics such as strength and ratios to be calculated for each segment.

**Negations**

By applying a negation detection algorithm, the following measurements related to the use of negation expressions can be extracted:

- Percentage of document terms affected by a negating expression.
- Total negating terms for a given document segment, assuming number of segments N, defined a priori.

The parts of speech being considered for SentiWordNet scores are adjectives, adverbs and verbs. According to information from *Table 11 - Scoring statistics per part of speech (Esuli et al, 2006)*, nouns are mostly objective and carry little opinion bias and will be left out of the final set of features. Further evidence suggesting this can also be seen on the classification performance results obtained in (Yu et al, 2003) where the best feature set taking parts of speech into account disregards the use of nouns. From *Table 7* it can be seen that most adverbs carry positive bias, hence an assumption will be made that positive adverbs are widely present on most texts and are redundant for the purposes of opinion classification, and therefore only negative adverbs will be included on the feature set. Adjectives, on the other hand have the potential to carry considerable opinion information, and therefore the feature set will be extended for this particular part of speech.

The table below details individual features to be extracted from a film review using SentiWordNet. The total number of features generated varies according to the number *N* of segments the document is divided into.

| Feature Type | Description of Features |
|---|---|
| Overall Scores per part of speech. | Sum of all positive scores for adjective.<br>Sum of all negative scores for adjective.<br>Sum of all positive scores for verbs.<br>Sum of all negative scores for verbs.<br>Sum of all negative scores for adverbs. |
| Scores *Strength* per part of speech. | Negative and positive strengths for adjectives.<br>Negative and positive strengths for verbs.<br>Negative strength for adverbs. |
| Ratios per part of speech. | Negative and positive ratio for adjectives.<br>Negative and positive ratio for verbs.<br>Negative ratio for adverbs. |
| Scores per document segment. | For a N-segmentation of document:<br>N positive scores for adjectives.<br>N negative scores for adjectives.<br>N positive scores for verbs.<br>N negative scores for verbs.<br>N negative scores for adverbs.<br>N sums of positive adjectives.<br>N sums of negative adjectives.<br>N positive scores for adjective strength.<br>N negative scores for adjective strength. |
| Negation | Percentage of negated terms in document.<br>Negated terms per document segment (for N segments) |

**Table 16 - SentiWordNet Feature Description**

**Parameters for Feature Generation**

The end result of the above set of feature is dependant upon how the extraction algorithm is configured. In this section, the key parameters affecting feature values extracted from SentiWordNet are detailed. Tuning of SentiWordNet feature generation parameters may affect the performance of the opinion classification task, and is included as an activity on the experiment described on the next chapters. The table below details the identified parameters.

| Parameter | Description |
|---|---|
| Scoring function | Determine score value as a function of term location within a document. Example scoring functions are:<br>• Linear growth with document position.<br>• Polynomial growth with document position. |
| Negation detection | Negation detection algorithm can be enabled or disabled. When enabled, a window size for terms to be negated can be specified. |
| Number of segments | Choice of value of N when partitioning the document into N separate segments. |
| Threshold for score consideration | Threshold value in positive/negative score differences when considering ambiguous synsets (See discussion on Section 5.4.2). |

**Table 17 - Parameters for Feature Generation**

## *5.6. Conclusion*

In this chapter the structure of the SentiWordNet lexical database of term opinion scores (Esuli et al, 2006), and the Polarity data set of film reviews (Pang et al, 2004) were analysed in more details, with the objective of determining how to best use SentiWordNet to build a model that represent opinion information from text documents. The analysis highlighted the need to avail of natural language processing techniques such as part-of-speech tagging to enrich the model, as well as potential limitations of using lexical resources for determining opinion information.

Lexical resources such as SentiWordNet contain opinion bias scores based on individual terms, and when building a model based on this type of information there are certain challenges stemming from the nature of natural language to be considered, as surveyed in Chapter 4. Word sense disambiguation becomes relevant, since terms with potentially multiple meanings may carry different opinion bias depending on context and their use within a sentence. Problems with a sentence's level of objectiveness also arise, when scoring terms that do carry opinion bias, but not strictly

related to author opinion, such as text extracts that objectively describe a film plot. Another challenge arises when negation sentences occur, potentially inverting the meaning and associated scores for a term. Domain-specific terms are also an issue, since they may indicate a different bias than that of their more commonly seen uses. The above issues naturally impose limitations to the effectiveness of sentiment classification using SentiWordNet. These were addressed with mitigating strategies where feasible, or acknowledged as relevant topics of research for further improving in the model.

The outcome of this chapter is the specification of a data model that reflects opinion information derived from SentiWordNet, and a proposed process for obtaining the features having the original Polarity data set as a starting point. From this specification, the process can be implemented with the assistance of third party tools for part-of-speech tagging, and scripting code for the generation of SentiWordNet features.

In the next chapters, the experiment on opinion mining will be described in details, and results will be presented and discussed. The experiment uses a data mining classification technique for determining opinion orientation of documents from the polarity data set based on SentiWordNet information, and the input to the classification task will be the set of features described by the specifications from this chapter. Additionally, the analysis of results of the classification experiment will take into consideration findings from the assessment of SentiWordNet capabilities and limitations conducted during the process of devising the data model.

# 6. SENTIMENT CLASSIFICATION EXPERIMENT

## *6.1. Introduction*

The use of lexical resources for performing opinion mining tasks has received considerable attention in the research literature and several approaches have been proposed, as surveyed previously on section 4.2.5. The underlying motivation for these techniques is the assumption that individual terms in a document are objects that carry unit opinion bias, which could in principle be used as a measure of opinion of the text document they belong to (Kim et al, 2004). Lexical resources that relate words to sentiment can be constructed manually by eliciting positive and negative words, or via induction methods on existing lexicons such as WordNet. SentiWordNet is a lexical resource built upon such approach via the generation of opinion scores on WordNet synsets from a core of paradigmatic words, using a semi-supervised machine learning classification process (Esuli et al, 2006). One interesting problem in opinion mining where SentiWordNet can be of assistance is sentiment classification: determining positive or negative opinion for a given document on a specific topic.

In Chapter 5, the first part of this dissertation's experiment was discussed: the SentiWordNet database was studied in details, and a set of features that extract sentiment information from documents using SentiWordNet was proposed. Using the polarity data set of film reviews comprising 2000 documents equally categorised into positive "thumbs-up" and negative "thumbs-down", this chapter presents a classification experiment that aims at determining how well can the SentiWordNet set of features perform sentiment classification, and how does it compare with other methods in the literature is the key motivation of this experiment.

In this chapter the experiment in sentiment classification with SentiWordNet is described in details. The experiment's objective, scope and approach are presented, followed by a detailed description of the experiment setup, key evaluation metrics and execution steps.

## *6.2. Objectives and Scope*

The experiment described on this section aims at assessing the viability of using SentiWordNet as an approach for performing sentiment classification on text. To further detail the line of investigation, the experiment objectives can be described as follows.

- Determine what classification performance can be achieved by using SentiWordNet features from Chapter 5, and how it compares to other results published in the literature.

- On Chapter 5 a proposed data set with features reflecting several aspects of opinion information that can be extracted with SentiWordNet was developed. This data set relies on a number of parameters that affect how features are computed. The experiment results should determine the effect of each individual parameter to overall classification performance, thus indicating how relevant each parameter is to sentiment classification.

- Evaluating the effect of increasing the number of training examples to classification performance, in comparison to a baseline classification method.

- Assess the training time and runtime characteristics of a classifier based on SentiWordNet, and compare it to a baseline classification method.

With the above experiments, it will be possible to evaluate how accurate, how fast and how robust can the SentiWordNet approach perform document-level sentiment classification. These indicators will give further insights into SentiWordNet's applicability to opinion mining problems in practical applications, and identify areas in sentiment lexicons and document-based classification with potential for development of better results. To achieve these goals, a labelled data set based on SentiWordNet information will be generated using the specifications derived in Chapter 5, and used as input to train a classifier that performs sentiment classification.

The tasks that need to be performed for the completion of the experiment are outlined in the milestones below.

1. Implement algorithm for extracting features from text documents based on SentiWordNet, as described in Chapter 5, using the polarity data set of film reviews as input (Pang et al, 2002).

2. Train a baseline classifier for sentiment classification based on unigram bag-of-words method similar to the one described in (Pang et al, 2002).

3. Evaluate results on classification performance using SentiWordNet feature set when applied to three standard classification algorithms: Naïve Bayes, Support Vector Machines and k-Nearest Neighbours.

4. Evaluate the effects of feature changes to SentiWordNet feature parameters (as described in Section 5.4) to classification performance.

5. Evaluate the effects of feature selection and outlier removal to classification performance.

6. Compare results obtained using SentiWordNet approach for opinion classification to the baseline classifier using to a set of evaluation criteria.

The above milestones reflect the major activities required by the experiment. First, an algorithm that extracts SentiWordNet features from plain text is needed to generate the experiment data set. Another requirement is the implementation of the baseline classifier where SentiWordNet results can be compared to. Sentiment classification tests with SentiWordNet are then performed by steps 3 to 6. In the following section, these milestones are logically grouped into execution stages, to be concluded in the order outlined above.

### 6.2.1. Out of Scope

To perform sentiment classification, this experiment leverages supervised learning algorithms from the data mining literature. There is a vast array of classification methods available and new methods being constantly developed, as discussed in 3.3.7. The experiment results are primarily interested in evaluating SentiWordNet, and whereas potentially better results can be obtained by testing a wider range of algorithms, it is not the objective of this dissertation to perform this comparison. Instead, a classification algorithm that provides good results based on a comparison between three methods, and which are commonly applied to text mining will be chosen and applied to sentiment classification tests using SentiWordNet.

As seen on section 3.3.2, supervised learning algorithms also rely on parameters that determine how classification models are built, often with a large number of possible combinations. Finding algorithm parameters that best suit the particular experiment being performed here is not considered within scope of this investigation, as it will focus instead on the effects of SentiWordNet to sentiment classification. It is acknowledged however that gains to classification performance metrics could be obtained by fine tuning the algorithm, further considerations on this topic will be made on experiment conclusions on chapter 8.

## *6.3. Experiment Process*

In order to achieve the desired objectives for the opinion mining experiment, and to ensure the experiment is consistently repeatable, an execution process is presented below. The process is logically subdivided into stages performing distinct, self-contained tasks of the experiment process. The below diagram illustrates the relationships between each stage, and activities within each stage are detailed in the next sub sections.

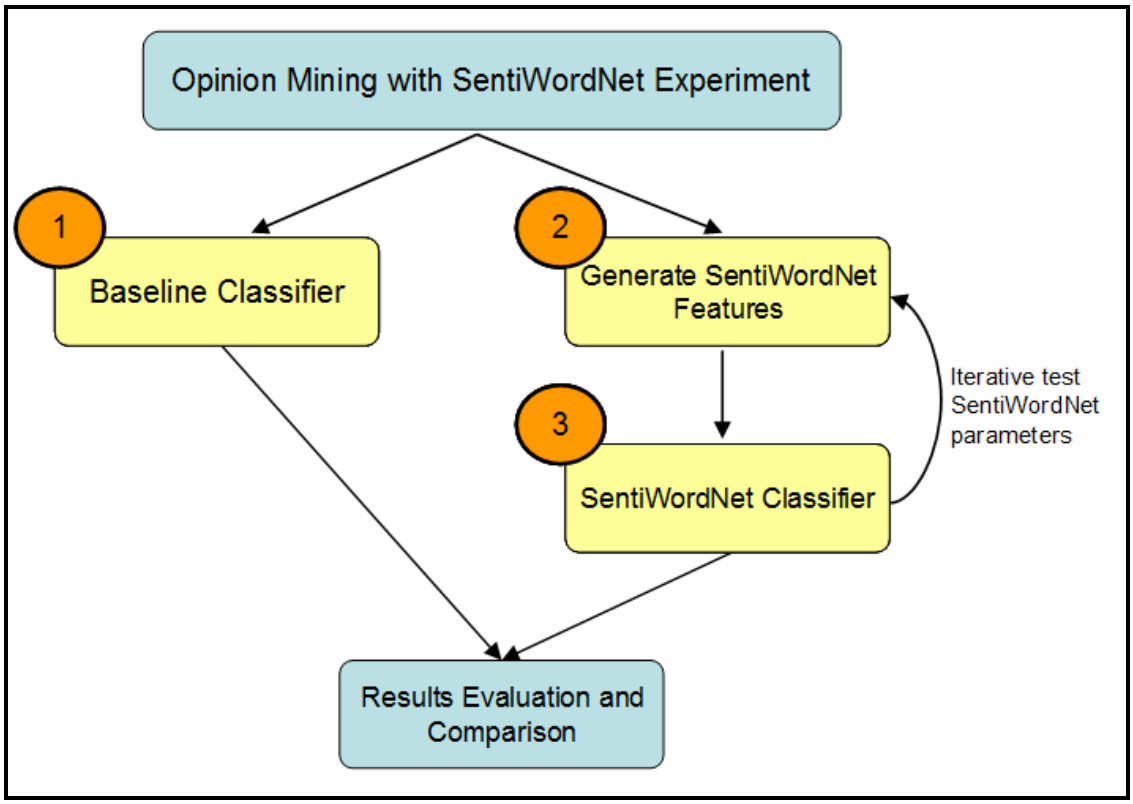



**Figure 12 - Experiment Execution Stages**

### 6.3.1. Baseline Classifier

For measuring the effectiveness of opinion mining with SentiWordNet, it would be useful to compare it with results of other techniques in the literature. In particular, a *baseline* method can be chosen using for criteria the fact it has been widely applied in text classification research problems, is closely related to seminal research in the area of opinion mining, and where results exist for the data set being used in this experiment. Implementing a baseline classifier will provide the ability to obtain results that represent closely the ones available in the literature, and to execute different tests when required by the experiment. For example, when comparing results using different cross-validation settings and with varying training data set sizes.

With this in mind, the chosen baseline method for comparisons is analogous to the one described in (Pang et al, 2002), comprising a classification task on the polarity data set according to its opinion label, using word vectors as the input data set. This method is commonly referenced in the literature when comparing opinion classification

experiments using the polarity data set. The building of the baseline classifier comprises the following activities:

1) Create word vector feature set from Polarity data set plain text film reviews;
2) Train and execute classifier algorithm.

The parameters chosen to be used as baseline in this dissertation's experiment are based on the best results found on the sentiment classification experiments described on (Pang et al, 2002).

### 6.3.2. Generate SentiWordNet Features

This stage is responsible for generating a data set with features derived from SentiWordNet, taking the documents from the Polarity data set as input, as described in Chapter 5. The outcome of this stage is a data set ready to be used by a classifier algorithm. As shown on Section 5.4 of the previous chapter, there are several parameters to be tuned in relation to how documents are scored with SentiWordNet, and we wish to assess how each parameter may affect the end result of the experiment. Thus, the feature generation stage is iterative, requiring several executions to generate different data sets with the required combination of parameters. This approach is in line with the discussed knowledge discovery methodologies in Chapter 3, where iterating through stages – in our case data preparation and data mining - is recommended for results refinement; and with the views of performing data mining in a data oriented manner advocated in (Hand et al, 2001; Kulkarni et al, 1998; Weiss et al, 1998)

### 6.3.3. Sentiment Classification Using SentiWordNet

A classifier algorithm can now be trained and executed taking a data set containing features generated using SentiWordNet in the Polarity data set from the previous stage. This stage will be executed iteratively as parameters for SentiWordNet features are refined. Upon completion the following tests will have been executed:

1) Select classification algorithm that provides best performance using SentiWordNet features.

- To this end, three classifier algorithms will be evaluated with standard settings: the Naïve Bayes, k-Nearest Neighbour and Support Vector Machines algorithms will be tested.

- The best method will be chosen based on classification accuracy results.

2) With the classification algorithm chosen from the previous stage, evaluate the impact of changes in SentiWordNet parameters described in Section 5.5 to classification performance.

   - Evaluate the relevance of each parameter to sentiment classification according to performance criteria established for the experiment.

   - SentiWordNet parameters being tested are number of document segments, negation detection, choice of scoring function and scoring threshold value for ambiguous synsets.

3) Using the best combination of parameters found on step 2, test and document the impact of feature selection and outlier detection to classification performance.

4) Using results from steps 2) and 3), evaluate the effect of different training set sizes to classification performance, in comparison to the baseline classifier.

As the experiment progresses through each step, it obtains classification performance results and selects the best method as the starting point for the next step. As discussed on Section 6.2.1, step 1 selects the classification algorithm based on standard parameters and a choice of 3 well known methods. For step 2, as illustrated on *Table 17 - Parameters for Feature Generation,* there are a series of parameters involved in the generation of the SentiWordNet data set that may affect classification performance. In practical terms, it would be prohibitive to attempt to evaluate all possible combinations of parameter values, together with variations on classification algorithm and training set sizes. Instead, the proposed approach described above iterates through

each individual parameter, finding the best possible classification results out of a subset of pre-determined options, before selecting another parameter for evaluation, a method is similar to the *simple greedy heuristic* search described in (Hand et al, 2001). At this point is worth mentioning that other parameter search approaches have been proposed in the literature, addressing the issue of finding sub-optimal solutions, a case that may occur in the greedy search method although it has been acknowledged to yield good results in practice with a simple implementation framework. A survey of other search optimisation techniques can be found in (Hand et al, 2001).

After the experiment execution, all results obtained in this stage will be documented and analysed in the next chapter, and used as the basis for this dissertation's conclusions. Based on the described experiment above, the next section introduces the criteria for comparing results, and discusses the chosen metrics for evaluation.

## *6.4. Criteria for Comparisons*

The final objective of any classification method is to perform predictions on new instances of data from training data as accurately as possible. To assess the feasibility of the method it would then be natural to find out if the resulting classifiers can obtain acceptable classification error rates, according to a certain pre-determined error metric. In addition, evaluating training and execution speeds is also of importance for any classifier used in practical applications where computing time is a constraint, and is of primary importance on environments where predictions are required in near real time. Finally, comparing how results behave with limited training data is also important, due to the cost involved in obtaining and labelling training data sets. The approach to performing sentiment classification using SentiWordNet uses distinct pre-processing steps and feature generation, and measuring factors such as speed, classifier performance and sensitiveness to training data would provide useful information for assessing the overall usefulness of the method.

To evaluate the SentiWordNet approach to sentiment classification, the obtained results will be compared with the ones obtained from the baseline classifier, which represents as closely as possible results available in the literature for the same data set

employed in this experiment. The metrics chosen for comparison based on the factors discussed above are presented in more details in the following sections.

### 6.4.1. Classification Performance

The key measurement for classification performance will be **classification accuracy**, introduced in Section 3.3.8. The polarity data set used for this experiment contains an equal number of documents for positive and negative classes, and there is no a priori distinction in importance between the two classes for the purposes of classification precision and recall. Accuracy is thus a suitable and easily understandable single value metric for this type of data set.

In addition, most results reported in the literature performing sentiment classification using the polarity data set use classification accuracy as a metric, thus it is reasonable to present results on the same metric so comparisons can be made. For cross-validation experiments, the result is typically presented in average accuracy across all folds.

### 6.4.2. Training Data Set Size

Finding labelled training sets for classification tasks is a non trivial and often expensive undertaking. It is therefore important for a method to achieve the best possible results from as little training data as possible. For that reason, classifier sensitiveness to training data will also be evaluated in comparison to the baseline classifier. This will be measured by comparing accuracy results using different sizes of training data. Two tests will be performed:

- Evaluate classification accuracy using 3-fold cross-validation for fractions of training data available: 10%, 25%, 50%, 75% and 100%.

- Evaluate classification accuracy for various cross-validation folds: 3, 5, 10 and 100.

### 6.4.3. Dimensionality and Runtime

Finally, dimensionality of a data set can be a determining factor in the training and execution time of a classification algorithm. High dimensional models can be

prohibitive when applied to certain classification methods, and may hinder the usefulness of a given model on certain applications where execution performance is paramount. Whereas the accuracy of any classifier is important, one could think of situations where results of automatic classifiers are needed in a near real-time fashion (e.g. to process results obtained from a search engine, or for processing live incoming streams of text from a news feed). For this reason, some consideration will be given to training and execution times of each method.

## *6.5. Experiment Setup*

In this section the execution environment of the experiment is fully described. The main objective is to provide detailed documentation of the experiment setup, tools, relevant parameter settings and execution order for future reference, and to ensure the process can be replicated as accurately as possible.

### 6.5.1. Technical Resources

In this section, all the technical resources used as part of the experiment, including software and physical hardware are provided. The chosen package for executing data mining classification algorithms was RapidMiner (Mierswa et al, 2006). RapidMiner is an open source data mining package with an intuitive user interface aimed at the rapid prototyping of data mining processes, an approach that suits the needs of a data mining experiment. It implements a wide range of algorithms, including the Naïve Bayes, Nearest Neighbour and Support Vector Machines applied in this experiment, and also implements functionality for performing text mining tasks directly from source text documents, facilitating the process of obtaining results from plain text data sets such as the Polarity data set of film reviews.

Additional scripting and integration with the SentiWordNet database were written in Python language, and the part of speech tagger used is the freely available Stanford POS Tagger described in (Toutanova et al, 2000).

| Resource | Selected Product | References and Comments |
|---|---|---|
| Data Mining Package | RapidMiner Community Edition v4.1 | www.rapidminer.com (Mierswa et al, 2006) |
| Scripting and Text Processing Language | Python v2.5 | www.python.org (van Rossum et al, 2003) |
| Part of Speech Tagging | The Stanford POS Tagger | (Toutanova et al, 2000) |
| Operating System | Windows Vista SP1 | www.microsoft.com/windows 32-bit edition. |
| Hardware | Intel x86 T2390 1.86GHz (dual-core). 4Gb RAM | Computer manufacturer and model is *Dell Inspiron 1525*. |
| Data Set | Polarity Data Set v2.0 | (Pang et al, 2004), and (Pang et al, 2002). Available from Cornell University Natural Language Processing Research Labs (1) |

(1) http://www.cs.cornell.edu/People/pabo/movie-review-data/

**Table 18 - Software and Hardware Resources**

### 6.5.2. Baseline Classifier

The baseline classifier performs sentiment classification using the Polarity data set using a machine learning method and word vector features, as described on (Pang et al, 2002). In this section this stage of the experiment is described in details.

**Word Vector Generation**

The first step in implementing the baseline is to convert textual documents from the Polarity data set into a word vector according to a word presence metric, such as the ones as described on Section **Error! Reference source not found.**. The following three methods were tested for the baseline implementation:

- TF-IDF Inverse term frequency measure.
- Binary: Accounts only for term presence in the document, possible values for any given term are present ( 1 ) or absent ( 0 ).
- Normalised Term Frequency within document.

Next, stop word removal and stemming are applied to reduce the total number of features. Stop words are commonly used terms, expected to be present on nearly all documents, and therefore of little value to detecting differences between them. The list of stop words used for the baseline classifier is detailed in the Appendix section A.1.

Stemming is a process of reducing a term to its root syntax. Applying stemming to documents as a pre-processing step tends to reduce the final number of features by converting variations of terms to a single representation. In this experiment Porter Stemming algorithm implemented by RapidMiner was employed.

**Classifier Algorithm**

The classifier algorithm trained on the word vector features is a Support Vector Machine using linear kernel. Results of the classification task will be measured using classification accuracy using 3-fold cross validation, as per the original experiment in (Pang et al, 2002). To ensure results are repeatable across all experiments, the sampling for generating each fold will use a fixed random seed. All the above term presence indicators will be tested using the classifier algorithm, and the one presenting best results will be chosen as the baseline classifier.

### 6.5.3. SentiWordNet Test Approach

The test approach for SentiWordNet will first select a classification algorithm based on results obtained on three commonly used methods, then iterate through different values

for each parameter described on *Table 17 - Parameters for Feature Generation*. In practical terms, it would be prohibitive to attempt to evaluate all possible combinations of parameter values, together with variations on classification algorithm and training set sizes. Instead, the proposed approach described below iterates through each individual parameter, finding the best possible classification results out of a subset of pre-determined options, before selecting another parameter for evaluation, a method is similar to the *simple greedy heuristic* search described in (Hand et al, 2001). The stages of the experiment's evaluation process are summarised on the diagram below and explained in the remainder of this section. On all stages, results will be evaluated using average accuracy over 3-fold cross validation.

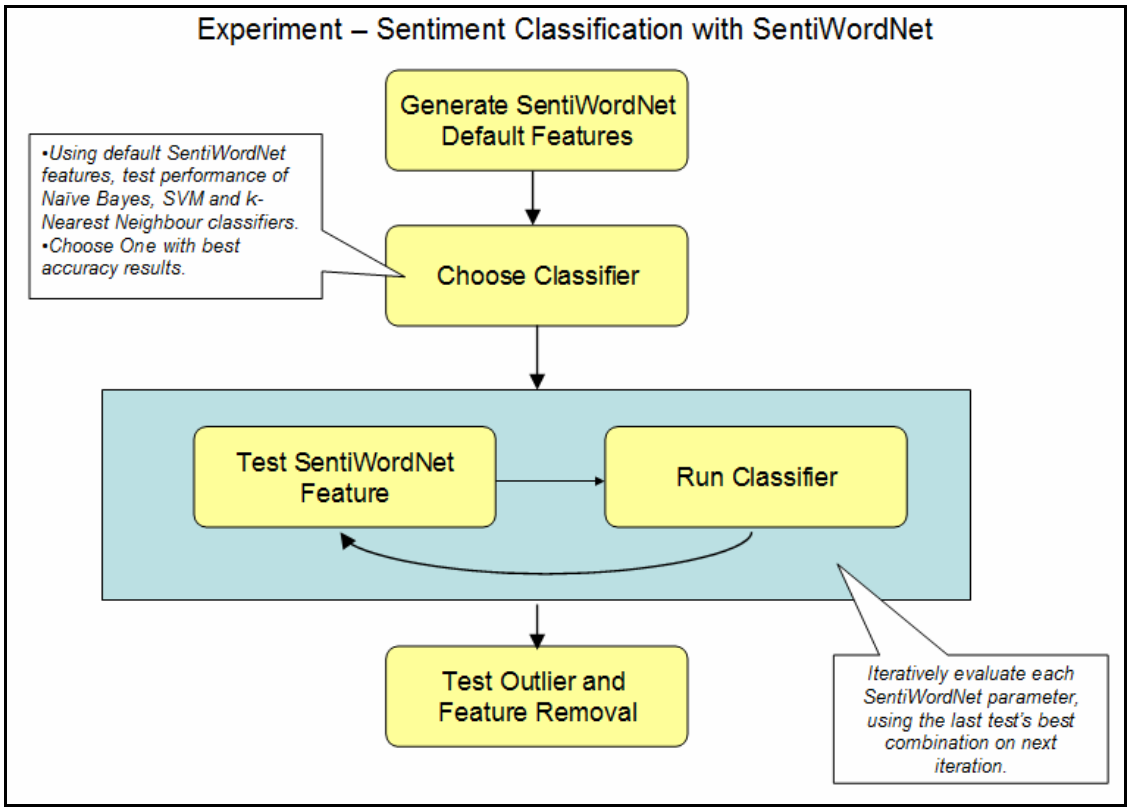



**Figure 13 - SentiWordNet Sentiment Classification Experiment**

### 1. Choosing Classification Algorithm

Using default parameter settings for generation of SentiWordNet scores, the classification performance of three classification algorithms will be evaluated. The objective of this stage is to select the algorithm to be used for the remainder of the

experiment, where SentiWordNet parameter changes will be evaluated. The table below details parameters used for each classification method.

| Algorithm | Classifier Parameter Settings |
|---|---|
| Naïve Bayes | Using Gaussian distribution for estimation of probabilities for real valued attributes. |
| k-Nearest Neighbours | Neighbours: k = 10<br>Distance Function: Euclidean Distance. |
| Support Vector Machine | Kernel Type: Linear<br>Error coefficient C: 1.0 |

**Table 19 -Classifiers and Parameters Tested in Experiment**

The above settings are fixed throughout the experiment, and as stated on section 6.1.1, tuning parameters for specific classification algorithms will not be attempted as part of this experiment. All results are evaluated using average classification accuracy using 3-fold cross validation, and in order to obtain repeatable results, a fixed random seed is used on all experiments

**Initial Parameters for SentiWordNet Features**

The default parameter settings for generating SWN features are described in the table below.

| Parameter | Default Value |
|---|---|
| Scoring Function | None |
| Scoring Threshold | 0.1 |
| Negation Detection | Disabled |
| Number of Document Segments | 5 |

**Table 20 - Default SentiWordNet Parameters**

The parameters above thus reflect an initial setup where no negation detection is being performed, and a scoring function is not being applied. This starting point was chosen so that the effects of testing such features in the next stages become more evident.

## 2. SentiWordNet Data Set

Once the classification algorithm has been selected from the previous stage, each individual parameter affecting the generation of the SentiWordNet data set is tested. The approach for testing the effects of each parameter reflects the data-driven approaches to data mining, widely prescribed in the literature (Hand et al, 2001; Kulkarni et al, 1998; Weiss et al, 1998), by iterative evaluation of the effects of parameter changes to classification performance. The table below details the tests and values for each parameter from the list detailed in Table 17. The process starts using the default values presented on Table 20 applied to algorithm with best classification results, selected from the previous stage, and progresses through each test, using the best results obtained up to that point as input to the next test. The table below details the values being tested for each parameter associated with SentiWordNet feature generation:

| Test | Evaluation | Parameters Tested |
|---|---|---|
| 1 | Scoring Functions | Linear Increase, Linear Decrease, Polynomial, None |
| 2 | Scoring Threshold | 0.5, 0.1, 0.05, 0.01, 0.001, 0 |
| 3 | Negation Detection (Window Size) | None (disabled), 1, 5, 10, 20 |
| 4 | Number of Document Segments | 5, 8, 10, 15 |

**Table 21 - Parameter Values Tested by Experiment**

On the initial test, the experiment evaluates various heuristics that calculate the strength of a given term in SentiWordNet as a function of its position in the document. This attempts to map stronger scores to areas of the document more likely to represent author opinion. The scores are calculated as a linear ascending and descending functions, polynomial increasing function of degree 2, and with no function thus only adding scores as they are found in text.

The second test evaluates the method for isolating SentiWordNet scores likely to be ambiguous. This is done by taking into account scores for a given term only when their positive and negative SentiWordNet scores differ by more than a given threshold, thus

assuming large score differences tend to indicate what opinion polarity is more likely to occur in general.

The third test evaluates the use of the detection algorithm for negated expressions, presented in section 5.4.2. Results are tested with the algorithm being disabled and for a series of values for the negation "window", or the number of terms where opinion scores will be inverted to the left or right, whenever a negating expression is found. We examine results for varying window sizes from a single term to 20, the average size of a sentence in the polarity data set. Finally, varying sizes for document segmentation are tested, from a starting number of 5 segments, to 15, which according to *Table 14- Polarity data set document statistics*, would represent an average of just over 2 sentences per segment.

**3. Feature Selection and Outlier Removal**

It is possible that the process that generates features the SentiWordNet data set creates redundant features that carry little information and do not assist in the separation of positive and negative film reviews. Also, since this data set originates from real-world data, it is very likely that noisy data, potentially misleading to a classification model could be present too. In other words, outliers may be present in the data set. Outlier detection and removal will be performed using a nearest neighbour method described in (Ramaswamy et al, 2000) and implemented in RapidMiner. The method works by finding data points with largest distance to the k-nearest neighbourhood they belong to, using the assumption that objects with a sparser neighbourhood than the majority of objects are likely to be outliers. The distance function can be adjusted, and Euclidean distance is being used for this experiment.

This stage of the experiment also performs an analysis of feature relevance to classification and feature selection based on the Chi-Squared correlation metric between attributes and the predicted class. The objectives of this task is to attempt to improve classification performance by eliminating uncorrelated, potentially noisy features from the data set, and also to provide insight into which features extracted from SentiWordNet provide little information for sentiment classification on this data set.

## *6.6. Conclusion*

This chapter provided a detailed description of the opinion mining experiment using the SentiWordNet database. The main objective of this experiment is to assess the use of SentiWordNet as a tool for document-level sentiment classification. The polarity data set of film reviews will be used as the source of subjective documents. This objective translated into a series of milestones, and subsequent tasks that compose the structure of the experiment. An experiment is proposed within a framework that ensures objectives are measured via key metrics, obtaining repeatable results is achieved through detailed documentation of execution steps, and results are comparable across other research in the literature.

Limitations of the experiment's approach were also outlined: since the key focus is on assessing SentiWordNet as a resource for sentiment classification, there is limited scope for fine tuning the parameters of classifier algorithms; for the same reason, the range of classification algorithms being evaluated by the experiment is not extensive.

The following chapter presents results obtained for all stages of the experiment, according to the metrics chosen for evaluation, discusses the method's strength and weaknesses, and proposes opportunities for further development arising from the information obtained from the experiment.

# 7. EXPERIMENT RESULTS

The previous chapter described an experiment that assesses SentiWordNet as a tool for performing sentiment classification according to performance measurements using the criteria of classification accuracy, training set sizes and runtime. This chapter presents the results obtained from the experiment execution and discusses them in the context of the experiment's objectives and key performance metrics.

## *7.1. Introduction*

This chapter presents results for the sentiment classification experiment using SentiWordNet, according to the experiment objectives and setup described in Chapter 6. The focus of this chapter will be in the presentation and analysis of results obtained for the key metrics established on section 6.4. The next section presents results for the baseline classifier and compares results with the ones obtained on the original experiment described on (Pang et al, 2002). Next, results for the SentiWordNet classification task are presented for each intermediate step of the experiment, as outlined in the discussion on SentiWordNet parameter test approach on section 6.5.3. Results are then presented for training and testing times. Lastly, the key results obtained from the experiment are presented in summary and discussed in light of other research on the area, examples of documents with inaccurate classification are explored, and the chapter is concluded with final remarks on the experiment and results.

## *7.2. Sentiment Classification Results*

In accordance with the experiment stages and key performance metrics discussed in section 6.4, results for the baseline classifier are presented, and results for the SentiWordNet classification experiment are presented for each intermediate step, with considerations on any improvements obtained. Finally, the best combination of parameters obtained with SentiWordNet are used for testing classification results with varying training set sizes, and results are presented in comparison to the baseline classifier.

### 7.2.1. Baseline Results

The baseline classifier is described in section 6.5.2. It comprises a support vector machine trained to perform sentiment classification using word vectors as features. Initially the process was tested with three distinct word vector representations: TF-IDF, presence and word frequency. The classifier was trained separately using each representation as input, using 3-fold cross-validation. Results for average accuracy on each type of word vector are presented below.

| **Word vector type** | **Accuracy (%)** |
|---|---|
| TF-IDF | 82.45 |
| Binary Presence | **83.90** |
| Word Frequency | 82.71 |

**Table 22 - Baseline Results and Word Vector Types**

The best accuracy result obtained is the case where word vector records only presence, and highlighted in bold face. This result is in accordance with observations in (Pang et al, 2002), where binary presence also obtained best classification results. The table below details accuracy, training time and runtime results obtained for varying cross-validation folds using binary presence as the word vector, and will be used for comparisons later in this chapter.

| **Folds** | **Accuracy** | **Training Set Size** | **Validation Set Size** | **Train Time** | **Execution Time** |
|---|---|---|---|---|---|
| 3 | 83.90 | 1333 | 667 | 31s | 29s |
| 5 | 83.65 | 1600 | 400 | 40s | 12s |
| 10 | 84.95 | 1800 | 200 | 48s | 10s |
| 100 | 84.45 | 1980 | 20 | 102s | 3s |

**Table 23 - Results for Baseline Classifier using Binary Presence Word Vector**

Training and execution times correspond to the average time recorded for RapidMiner to perform one fold. It should be noted that, as the number of folds increase, so does the training set size, whereas the validation set where predictions are to be made

diminishes in size. The results for 3-fold cross validation are comparable to the 82.9% obtained in (Pang et al, 2002), which is indicative of the classification performance achieved by the method. The difference could be attributed to choice of cross validation folds, algorithm implementation and parameters, and the choice of features. We also note that (Pang et al, 2002) uses all document unigrams as input resulting in a word vector with 16165 features, whereas the baseline classifier uses a total of 2012 features. The reduction is obtained by performing word stemming as a pre-processing step as described in section 6.5.2.

### 7.2.2. SentiWordNet Parameter Tests

This section presents the results for sentiment classification performed using SentiWordNet features, as detailed in section 5.5. The results are detailed for each individual experiment step, as described in section 6.3.3. These are:

1. Select classification algorithm.
2. SentiWordNet feature parameter testing.
3. Outlier removal and feature selection.

**1. Select Classification Algorithm**

The first step is to train and execute three distinct algorithms that perform sentiment classification using features derived from SentiWordNet, generated using the standard parameters described in *Table 20 - Default SentiWordNet Parameters*. The classifier with best results is then chosen for the following experiments on testing SentiWordNet parameters, outlier removal and feature selection. The table below presents results for average accuracy using 3-fold cross-validation for each algorithm.

| **Algorithm** | **Average Accuracy (%)** |
|---|---|
| k-Nearest Neighbours<br>(k=10; Euclidean Distance) | 60.20 |
| Naïve Bayes | 63.05 |
| Support Vector Machine<br>(Linear Kernel, C=1.0) | **67.40** |

**Table 24 - Results for SentiWordNet Features using 3 Classification Algorithms**

The above results indicate that Support Vector Machines perform the best with this feature set. On this basis, it will be chosen as the classification algorithm for the further refinements performed on the remaining of the experiment. It is worth remembering that the focus of the experiment lies on testing of SentiWordNet features as an approach to sentiment classification, and tuning of algorithm parameters or testing of individual algorithms is outside of the experiment scope, as outlined on section 6.2.1. Improvements in accuracy results could be obtained with further investigation on these mentioned areas.

**2. Parameter Testing for SentiWordNet Features**

With the support vector machine classification method chosen from the previous step, a series of tests on SentiWordNet parameters were performed. These tests aim at assessing the impact of groups of features representing a certain aspect of the data set built from SentiWordNet, as illustrated on section 5.5. The parameter tests were discussed during the experiment description in section 6.5.3, and are executed in the following order:

1. Scoring function
2. SentiWordNet scoring threshold
3. Negation Algorithm and Negation window size.
4. Number of document segments.

**Test 1 - Scoring Function**

In this first test, four types of scoring functions were used to adjust individual terms as a function of position in document. The remaining SentiWordNet parameters are set as per initial configuration described in section 6.5.3. The table below details results obtained for average accuracy using 3-fold cross-validation for each scoring function.

| **Scoring Function** | **Average Accuracy (%)** |
|---|---|
| None | 67.40 |
| Linear Increasing | **68.00** |
| Linear Decreasing | 67.25 |
| Polynomial | 67.50 |

**Table 25 - Accuracy Results for Varying Scoring Functions**

The results show adjusting scores according to a linearly increasing function of term position in the document gives the best results in this data set. It could indicate the correlation of author sentiment and term position is stronger towards the end of the document, where concluding remarks about a film are more likely to occur.

**Test 2 – SentiWordNet Scoring Threshold**

As a measure of the certainty of the SentiWordNet score for a given term, the scoring threshold determines what is the minimum difference between the positive and negative scores of a given term in order for the term score to taken into account. On Chapter 5, we have seen that a given term in SentiWordNet may have both positive and negative scores, and that a polysemous term (e.g. one containing more than one meaning) will have more than one SentiWordNet score also. Terms where scores differ by a large amount are assumed to be heavily positive or negative biased, and therefore likely to be less ambiguous. The table below presents results obtained for various threshold values, using the best results from the previous test as starting point. Again, results are for average classification accuracy using 3-fold cross-validation, with best results obtained highlighted in boldface.

| **Threshold Value** | **Average Accuracy (%)** |
|---|---|
| 0.5 | 61.60 |
| 0.1 | 68.00 |
| 0.05 | 67.70 |
| 0.01 | 68.10 |
| 0.001 | 68.05 |
| 0.0 | **68.25** |

**Table 26 - Accuracy Results for Varying Scoring Threshold Values**

The above results indicate there is no benefit in ignoring term scores according to differences in positive and negative scores as a heuristic for term disambiguation: the best results obtained used a threshold value of 0, effectively using all found terms from the calculation.

**Test 3 – Negation Detection and Window Size**

The negation detection algorithm is based on the work of (Chapman et al, 2001) and detailed in section 5.5. Its objective is to detect when term orientation is being affected by a negating expression. When a negating expression is found, the algorithm inverts term scores for terms ahead or before the negating expression, up to a maximum of terms specified by the window size. The table below presents average accuracy results for varying sizes of negating window and for the case where the negation algorithm was not used.

| **Negation Window** | **Accuracy (%)** |
|---|---|
| Off | 68.25 |
| 1 | 67.55 |
| **5** | **68.50** |
| 10 | 67.65 |
| 20 | 67.50 |

**Table 27 - Accuracy Results for Negation Algorithm with Varying Window Sizes**

In the above, a minor improvement was achieved by implementing negation algorithm using a window size of 5 terms, bringing the average accuracy result to 68.5%.

**Test 4 - Document Segmentation**

Finally, document segmentation was tested for varying number of segments, with the remaining parameters fixed at the best results obtained on each previous test. The table below presents results for average accuracy over 3-fold cross validation.

| **Number of Segments** | **Accuracy (%)** |
|---|---|
| 5 | 68.50 |
| 8 | 67.95 |
| 10 | **68.55** |
| 15 | 65.25 |

**Table 28 - Accuracy Results for Varying Number of Segments**

The best result is highlighted in boldface, with only minimal improvement over previous accuracy results, using 10 segments.

**Final Parameter Settings**

After completing the parameter tests using SentiWordNet presented above, the final parameter settings yielding accuracy results of **68.55%** are detailed below.

| **Parameter** | **Best Value** |
|---|---|
| Scoring Function | Linear Increasing |
| Scoring Threshold | 0 |
| Negation Detection | Enabled, Window = 5 |
| Number of Document Segments | 10 |

**Table 29 - Best SentiWordNet Parameters Obtained with Experiment**

### 7.2.3. Outlier and Feature Selection

The final stage of the experiment performs a refinement of results obtained thus far, by performing outlier removal and feature selection using the parameters described on Table 29, presented in the previous section.

**Outlier removal**

Outlier removal is performed by using a k-nearest neighbour algorithm for identifying outliers based on its relative distance to other data points, as described on (Ramaswamy et al, 2000). Euclidean distance was used as the algorithm's distance metric. The table below presents average accuracy results obtained by removing a varying number of outliers to be removed, and for two possible values of k neighbours.

| **Number of Outliers** | **Accuracy (%) k=5 neighbours** | **Accuracy (%) k=10 neighbors** |
|---|---|---|
| No Outlier Removal | 68.55 | 68.55 |
| 5 | 68.35 | 68.25 |
| 10 | 67.75 | 68.10 |

| | | |
|---|---|---|
| 25 | 67.80 | 67.85 |
| 50 | 68.20 | 67.95 |
| 100 | **68.95** | 68.80 |
| 150 | 68.40 | 68.90 |
| 200 | 68.45 | 68.30 |

**Table 30 - Accuracy Results with Outlier Removal**

For 3-fold cross validation, the best results obtained using a value of 5 nearest neighbours for inspection, and 100 outliers removed, which corresponds to approximately 7.5% of the total number of documents available in the training set.

**Feature Selection**

Feature selection was performed by progressively removing features that have the weakest correlation to the positive or negative label being trained. In essence, these are features that when observed individually, are less likely to separate between positive and negative document classes. To detect which features to remove, chi-squared correlation weight was extracted from the data set as a preparation step using RapisMiner's chi-square weighting operator. The results are presented below for the bottom 20 features generated from SentiWordNet using the best parameters obtained during the experiment, and detailed on Table 23. Scores represent the relative feature weights from the least to most correlated features.

| Position | Attribute Name | Description | Weight relative to Chi-Squared Correlation |
|---|---|---|---|
| 1 | negbin8 | Number of negations in segment No. 8. | 0 |
| 2 | advnegbin8 | Total score for adverbs with negative scores in document segment No. 8. | 0.002417466 |
| 3 | negbin9 | Number of negations in document segment No. 9. | 0.007077381 |
| 4 | anbin5 | Number of adjectives with negative scores in document segment No. 5. | 0.007351313 |
| 5 | posvpct | Percentage of positive verbs out of total verbs found. | 0.010475046 |
| 6 | advnegbin5 | Total score for adverbs with negative scores in document segment No. 5 | 0.013643333 |
| 7 | apbin10 | Number of adjectives with positive scores in document segment No. 10. | 0.017331827 |

| | | | |
|---|---|---|---|
| 8 | advnegbin9 | Total score for adverbs with negative scores in document segment No. 9. | 0.018631441 |
| 9 | negbin2 | Number of negations in segment No. 2. | 0.023946131 |
| 10 | nbin6 | Total score for adjectives with negative scores in document segment No. 6. | 0.024514132 |
| 11 | advnegbin7 | Total score for adverbs with negative scores in document segment No. 7. | 0.025177349 |
| 12 | nbin3 | Total score for adjectives with negative scores in document segment No. 3. | 0.028163759 |
| 13 | advnstre | Percentage of negative adverbs from total adverbs found. | 0.03081977 |
| 14 | advnegbin4 | Total score for adverbs with negative scores in document segment No. 4. | 0.030881908 |
| 15 | nbin2 | Total score for adjectives with negative scores in document segment No. 2. | 0.031037958 |
| 16 | nbin8 | Total score for adjectives with negative scores in document segment No. 8. | 0.032052898 |
| 17 | advnegbin2 | Total score for adverbs with negative scores in document segment No. 2. | 0.032631923 |
| 18 | advnegbin3 | Total score for adverbs with negative scores in document segment No. 3. | 0.033570132 |
| 19 | anbin10 | Number of adjectives with negative scores in document segment No. 10. | 0.033686192 |
| 20 | anbin3 | Number of adjectives with negative scores in document segment No. 3. | 0.033822422 |

**Table 31 - 20 SentiWordNet Features with Lowest Correlation to Label Using Chi-Squared Test**

We can now test the effects in accuracy of removing features with low correlation to the document. The table below presents results for average accuracy using 3-fold cross-validation and by progressively removing features from the data set, according to the list from the above table.

| **N Least Correlated Features Removed** | **Accuracy (%)** |
|---|---|
| 3 | 68.65 |
| **5** | **69.10** |
| 10 | 68.15 |
| 15 | 68.15 |
| 20 | 67.95 |

**Table 32 - Accuracy Results with Feature Removal**

Using to this method, optimal results are obtained when the 5 least correlated features are removed from the training process, resulting on average accuracies of **69.10%**. Further removals generate loss of information on the model, affecting accuracy results.

### 7.2.4. Training Data Set Size and Execution Time

Upon execution of the previous experiments, we now turn our attention to the effects of sensitivity to training data set size and runtime execution. The table below presents results for the best parameter combination for SentiWordNet, outlier removal and feature removal from the previous steps, for various sizes of cross-validation folds.

| **Folds** | **Average Accuracy** | **Training Set Size** | **Validation Set Size** | **Train Time** | **Execution Time** |
|---|---|---|---|---|---|
| 3 | 69.10 | 1333 | 667 | 65s | 1.6s |
| 5 | 67.75 | 1600 | 400 | 92s | < 1s |
| 10 | 69.30 | 1800 | 200 | 135s | < 1s |
| 100 | 69.05 | 1980 | 20 | 156s | < 1s |

**Table 33 -Accuracy and Experiment Timings with Outlier Detection**

Training time refers to time taken in seconds to train the predictive model using a Support Vector Machine classifier based on the training set size given. Execution time is the time in seconds taken to perform all predictions on the validation set. From the above results, a clear improvement can already be noted on *execution time* of the algorithm, due to the reduced size of the feature set. On early tests it was noticed that outlier detection step had a large contribution to overall training time. To make more accurate training time comparisons with the baseline classifier, which does not include this step, the next table presents results for SentiWordNet for various cross-validation folds without the outlier detection and removal step.

| Folds | Average Accuracy | Training Set Size | Validation Set Size | Train Time | Execution Time |
|---|---|---|---|---|---|
| 3 | 68.60 | 1333 | 667 | 35s | < 1s |
| 5 | 68.65 | 1600 | 400 | 44s | < 1s |
| 10 | 68.90 | 1800 | 200 | 59s | < 1s |
| 100 | 69.00 | 1980 | 20 | 86s | < 1s |

**Table 34 - Accuracy and Experiment Timings without Outlier Detection**

**Reduced Training Size**

To verify the effect on reduced training set sizes to the overall classification accuracy, the same training process was repeated, this time using only a fraction of the original training set sizes for both the baseline method, and SentiWordNet using the optimal parameters for feature generation reported on Table 29, outlier removal and removed features obtained from Section 7.2.3. The number of outliers removed was adjusted proportionally to the percentage of original data available for training. The results are presented in the table below using 3-fold cross-validation.

| % Of Original Training Set | Average Accuracy (%) - SentiWordNet | Average Accuracy (%) - Baseline |
|---|---|---|
| 10 | 55.75 | 71.89 |
| 25 | 61.41 | 79.92 |
| 50 | 63.59 | 82.04 |
| 75 | 65.77 | 82.82 |
| 100 | 69.1 | 83.9 |

**Table 35 - Accuracies for Various Training Set Sizes**

## *7.3. Result Analysis and Considerations*

This section presents key results obtained from the SentiWordNet classification experiment, with comparisons to a well know baseline sentiment classification method described in (Pang et al, 2002) and implemented as part of this experiment. A

discussion on misclassified entries using the SentiWordNet approach is also presented, with a view of highlighting further possible improvements to the method.

### 7.3.1. Accuracy Results

In this section the accuracy results for the three stages of the experiment are analysed in more details. The table below details the improvements in classification accuracy obtained at each step testing SentiWordNet feature parameters, when running the experiment with 3-fold cross validation.

| **Stage** | **Best Result** | **Accuracy (%)** |
|---|---|---|
| Choose Classifier | Initial Support Vector Machine Classifier | **67.40** |
| SentiWordNet Parameters | Linear Scoring Function | 68.00 |
| | Scoring Threshold = 0 | 68.25 |
| | Negation Window = 5 | 68.50 |
| | Segment Size = 10 | **68.55** |
| Outlier and Feature Removal | Outlier Removal | 68.95 |
| | Feature Selection using Chi-Square Correlation | **69.10** |

**Table 36 - SentiWordNet Sentiment Classification Results**

**Choice of Classifier**

Initially, it can be observed from the above that the Support Vector Machine classification technique yielded the best classification accuracy amongst the three methods compared. The positive results of Support Vector Machines reflect similar outcomes obtained on classifier comparisons in the literature for text mining (Joachins, 1998), and opinion mining (Pang et al, 2002).

**SentiWordNet Features**

On SentiWordNet parameter testing, the best performing scoring function weighted term scores increasing linearly as a function of its position in the document, improving accuracy to 68.00%. As noted in (Pang et al, 2002) and discussed previously in section 5.4.2, this suggest intuitively that the way a typical review is structured may position

opinion information more heavily towards the end of the document, where an author's final concluding remarks are commonly placed. It is interesting however to note also that scoring more heavily the final terms of a review, such as when using the polynomial function did not yield better results than the linear case, suggesting opinion information also exist on other parts of the document and should be taken into account as well.

The next parameter evaluated was the scoring threshold used to discard terms with potentially ambiguous scoring in SentiWordNet. The experiment reveals that there was no gain in predictive accuracy when discarding terms where positive and negative scores exist and are closer to each other by more than a specified threshold value. In other words, this heuristic caused the loss of opinion information for some terms, reflected in the accuracy results presented in *Table 20*, and the best approach was to simply include scores for all found terms, regardless of their potentially ambiguous bias, yielding an average accuracy result of 68.25%.

The negation algorithm employed for detecting negating expressions and adjusting scores in SentiWordNet was tested with several sizes of negation windows, and only a minor improvement was obtained for the case where a negating window size of 5 terms was used, bringing average accuracy to 68.50%. The use of negation is however, closely linked to writing style and this small improvement suggests more development on highlighting opinion expression from negations could be possible.

Finally, to further investigate the effect of individual parts of a review to the overall review sentiment, documents were divided into segments, and SentiWordNet scores was calculated for terms belonging to each segment separately. The best results for this step of the experiment were obtained from dividing documents into 10 segments, but only marginally differed from accuracy obtained using only 5 segments. Using results from *Table 14- Polarity data set document statistics*, each of the 10 segments would contain on average 3.5 sentences, and 61 to 68 terms.

**Outliers**

Removing outliers from the training data set by employing a k-Nearest Neighbour technique further improved accuracy results for 3-fold cross-validation to 68.95%,

suggesting the existence of reviews on the Polarity data set that damage the accuracy of the SentiWordNet predictive model when used on training. As shown on the next section, however, this result is less relevant as the number of training samples increases, suggesting the classifier algorithm is able to adjust to a better model with the increase in training data.

**Feature Removal**

The Chi-Squared test for correlation applied to the features extracted from SentiWordNet measures the relationship between each individual feature and the positive or negative classes of reviews. Lower correlation values indicate that a given feature value is likely to occur independently of the positive or negative classes of reviews, and therefore unlikely to assist in class prediction. Removing the 5 features with lowest correlation scores from the data set improved accuracy results to 69.10%.

Amongst the 5 least correlated features shown in Table 25, low correlation between the document class and features that measure the number of negated terms for a document segment (*negbin* features) can be seen, suggesting the use of negating expressions in the narrative of a review is equally likely to be used on both positive and negative instances. Low correlation values also appear on several features measuring negative SentiWordNet scores for adverbs in a given document segment (*advnegbin* features), and so is the feature measuring the percentage of verbs with positive scores out of total verbs found in the document (*posvpct*). Indeed, this last feature has a value of zero for most documents, indicating as pointed out on *Table 11 - Scoring statistics per part of speech (Esuli et al, 2006)*, that SentiWordNet scores tend not to carry opinion polarity for the majority of verbs.

### 7.3.2. Baseline Comparisons

This section compares results obtained from the experiment using SentiWordNet with the baseline classifier for the key metrics outlined in Chapter 6. The results presented here will also help this dissertation's concluding discussion on the SentiWordNet approach for sentiment classification, and further research directions.

**Classification Accuracy**

We begin by presenting classification accuracy across different sizes of cross-validation folds for SentiWordNet and baseline methods.

| Folds | Baseline | SentiWordNet with Outlier | SentiWordNet without Outlier |
|---|---|---|---|
| 3 | 83.90 | 69.10 | 68.60 |
| 5 | 83.65 | 67.75 | 68.65 |
| 10 | 84.95 | 69.30 | 68.90 |
| 100 | 84.45 | 69.05 | 69.00 |

**Table 37 - SentiWordNet and Baseline Classification Accuracy**

The graph below illustrates the three results.

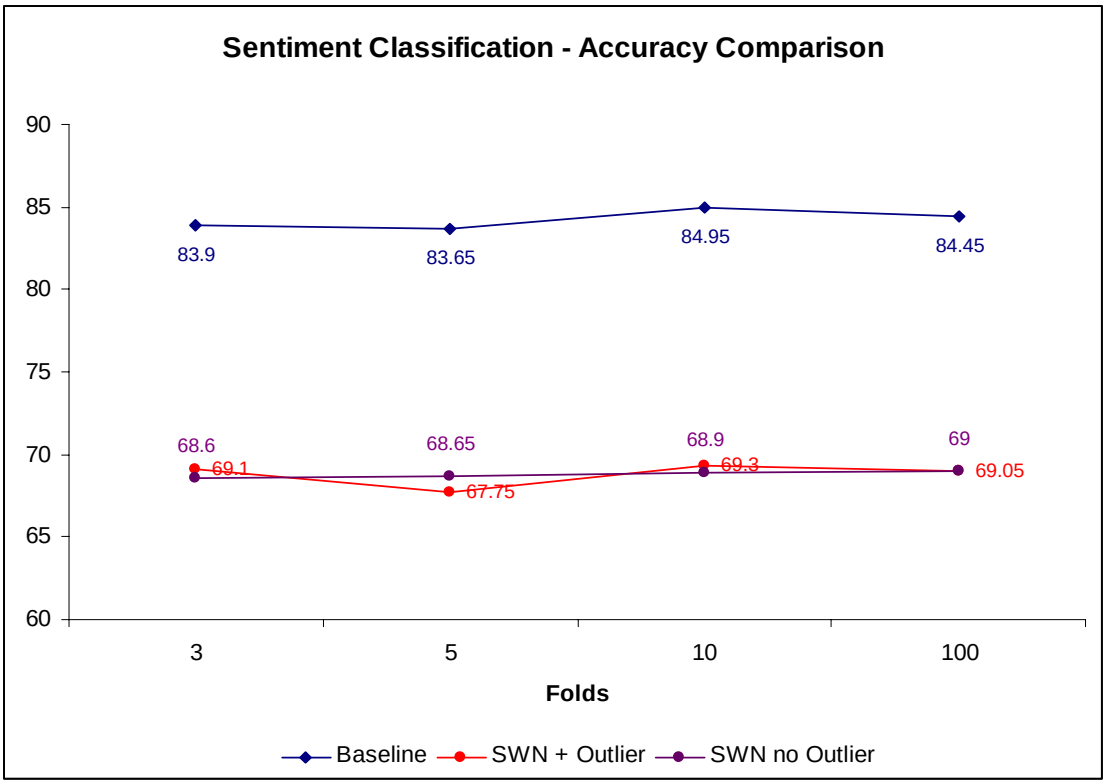



**Figure 14 - Accuracy Comparisons for various Cross-Validation Folds**

All accuracy results for the SentiWordNet experiments remained well below the baseline classifier using unigrams. On the experiments using a higher number of folds, classification accuracy for SentiWordNet without the outlier removal step improves and the results are comparable to the ones obtained when this step is used, indicating a

more robust model as the training data set increases minimising the need for outlier removal, or that outlier detection parameters should be tuned according to training set sizes for better results.

The results obtained however are close to the 69% obtained in (Pang et al, 2002) using a classifier based on a list of positive and negative words purpose built for the movie domain, and document statistics. This may indicate SentiWordNet is capable of achieving similar accuracy results for this class of technique, however with the difference of not being tailored towards a specific domain, but rather using a generic set of term scores derived from SentiWordNet.

**Training Set Sizes**

To complement the analysis of accuracy results with various sizes of data sets, the table below illustrates results obtained with different fractions of the original data set available for training. All results were tested using 3-fold cross-validation.

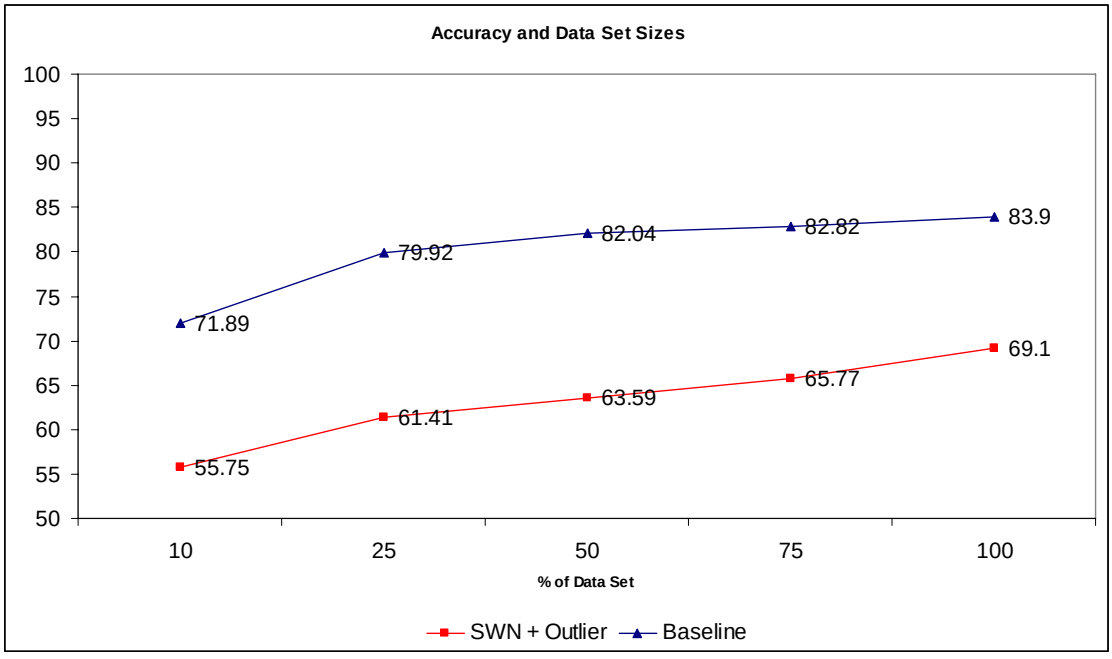



**Figure 15 - 3-Fold Accuracies for Different Training Set Sizes**

As before, SentiWordNet accuracies remain below the baseline classifier, but as the two linear trends suggest, the SentiWordNet method benefits more from more training data available, narrowing the difference between the two methods slightly as the training data set size increases.

**Training Time and Execution Time**

The execution time of the SentiWordNet classifiers stayed within less than 2 seconds on all experiments, whereas the baseline classifier required higher execution times, as outlined on the table below.

| Folds | Validation Set Size | Baseline | SentiWordNet with Outlier | SentiWordNet without Outlier |
|---|---|---|---|---|
| 3 | 667 | 29s | 1.6s | < 1s |
| 5 | 400 | 12s | < 1s | < 1s |
| 10 | 200 | 10s | < 1s | < 1s |
| 100 | 20 | 3s | < 1s | < 1s |

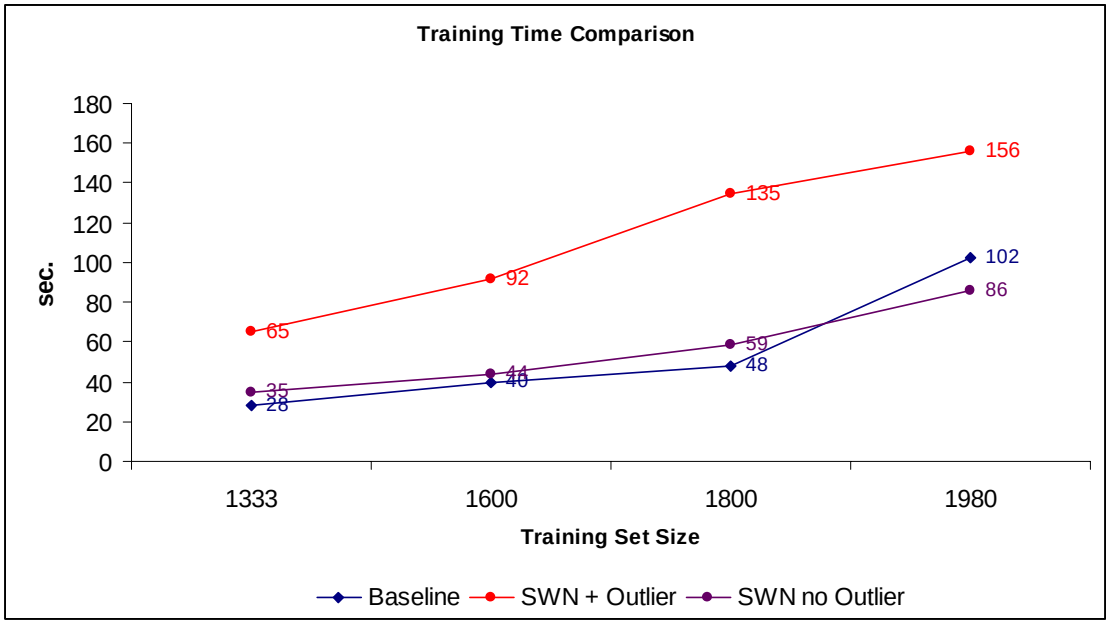
**Table 38 - Execution Times for Baseline and SentiWordNet Classifiers**

Results are calculated by averaging execution time measured over the experiment folds. As stated earlier, a higher number of cross-validation folds increases the training set size, while reducing the validation set size, hence the decreasing execution times for higher folds. We can calculate from the above results that, when applying the computing resources and algorithm implementation used on this experiment, the baseline classifier performs a prediction on average in **0.04s** for the case with 3-fold cross-validation. Due to timing precision of the measurements it is not possible to determine a similar result for the SentiWordNet classifier, but an upper limit to individual prediction time would be **0.003s**, when execution time is exactly 1 second. The difference between the baseline classifier and SentiWordNet can be attributed to the low dimensionality of the SentiWordNet model: SentiWordNet performs predictions based on 80 features, whereas the baseline classifier needs 2012 features. As seen in Chapter 3, the performance of support vector machines does not necessarily correlate to the number of dimensions, but to the number of support vectors. Thus the SentiWordNet model requires a much smaller number of support vectors to perform the classification task than the baseline.

For training times, the table and graph below summarise times in seconds for the baseline and SentiWordNet classifiers using different cross-validation folds.

| Folds | Training Set Size | Baseline | SentiWordNet with Outlier | SentiWordNet without Outlier |
|---|---|---|---|---|
| 3 | 1333 | 31s | 65s | 35s |
| 5 | 1600 | 40s | 92s | 44s |
| 10 | 1800 | 48s | 135s | 59s |
| 100 | 1980 | 102s | 156s | 86s |

**Table 39 - Training Times for Baseline and SentiWordNet Classifiers**




In absolute terms, it is natural to expect training and execution times for any classification algorithm to depend on external factors such as processor speed, available memory, programming language and implementation details. However by comparing results obtained from SentiWordNet and the baseline classifiers using the same algorithm implementation and data mining tool, it can be seen the low dimensionality nature of the SentiWordNet model does not strongly affect training times, when compared to a model with high number of features such as the unigram bag-of-words method of the baseline classifier. This is in line with the expected theoretical behaviour of support vector machine training seen in Chapter 3, which

states that training time is a function of the size of the training set, but not of dimensionality. Interestingly, the training times for the baseline classifier outperform SentiWordNet on nearly all training set sizes, even for the case with no outlier detection. It can be speculated that implementation details of Support Vector Machines in RapidMiner, and the nature of features on the bag-of-words method, which only admits values of 0 or 1, are likely causes for the speedup on the training process.

### 7.3.3. Analysis of Misclassifications

This section presents a closer examination at film reviews incorrectly classified by the SentiWordNet approach. This will assist us in evaluating the method's weaknesses and draw some conclusions and propose possible improvements to the method. The following are text extracts from a two film reviews with an overall negative sentiment where the SentiWordNet classifier made incorrect predictions. The extracts were taken from the concluding remarks at the end of each document:

> *"wild wild west's bright spots , such as the cool opening credits sequence , bai ling's all-too-brief appearance as a femme fatale , or the brilliant " his master's voice " joke , are all part of the film's first half , which is more clever and enjoyable , at least , than its second ."*

> *"summer of sam has some superficial elements of a good film : it looks great , it has a few notable performances and i suppose it's pretty well directed , in a purely technical way ."*

The first aspect to be noticed relates to the order of opinions presented, also described as the phenomenon of *thwarted expectations*, already noted in (Pang et al, 2008) and highlighted as a limiting factor to bag-of-words classifiers such as the baseline implementation of this experiment (Pang et al, 2002). In such cases the author builds up an expectation by describing positive aspects of the movie, only to later frustrate it by presenting another negative aspect. From the above, it can be seen a rich amount of words denoting positive bias (e.g. "looks great", "notable performances", etc) that would be heavily weighted in the SentiWordNet model, since they appear at the end of the document, but which however do not contribute to overall film sentiment. It would

appear that methods that rely solely on term polarity would not perform well for this type of narrative, and richer methods for analysing discourse would be necessary.

Another aspect to be noted is that relying only on term orientation would include irrelevant words, such as film titles (e.g. “wild wild west”), actors and place names whose terms may carry positive or negative term orientations and affect document scoring. Cases like these could be improved by introducing named entity recognition as a pre-processing step to detect and ignore such terms where appropriate. One example of this step applied to sentiment classification can be seen in (Dave et al, 2003).

The two above reviews present another a similar characteristic: they are relatively long texts, richly describing film features, scenes and plots. The longer the review, the more likely it is to also include off-topic sections, such as character dialogues or descriptions of a different film for comparison. According to SentiWordNet model, terms involved on purely descriptive sections of the document would also be accounted for, if SentiWordNet scores exist for them, thus potentially causing problems to overall scores and classifier predictions. One interesting path to address this issue is to implement a subjectivity detector in an attempt to filter out sections with predominantly descriptive content. Subjectivity detection was discussed in section 4.2.3 of Chapter 4, and an experiment using the polarity data set is described in (Pang et al, 2004). It is worth highlighting however that this task is not a trivial one, and even descriptive sections are likely to contain important but subtle clues about author opinion, as in the example below.

> *“… the mad inventor who is plotting to divvy up the united states and sell it back to britain and spain . how will loveless accomplish this ? well , by hulking around the desert in an enormous , mechanical tarantula , of course .”*

Here, the author describes the plot but also introduces ironical remarks by way of choice of words and style, which could be lost if the sentence were removed by a subjectivity detection step.

As noticed in (Pang et al, 2008), reviews are likely to differ considerably in style, and choice of vocabulary. Consider the extract below from a misclassified example with negative overall sentiment:

*“ the film is just a reel to show off a bunch of snazzy fx shots . “*

The style of the above review is more colloquial, uses shortened words - the term “fx” for special effects - and expressions like “show off”, which are harder to detect: in the experiment’s case, only the term “show” appears as a verb after part of speech tagging. It suggests that some opinion information may be lost on cases like the above, which could be retrieved by introducing text pre-processing steps to facilitate the detection of expressions, and an extended lexicon to improve understanding of colloquial language.

From the same document, another example of incorrect part-of-speech tag can be found on the use of the word “asteroid”. The tagger program classified the term as an adjective, for which a SentiWordNet score exists: according to WordNet, the term “asteroid” not only describes a celestial object, but is a also an adjective, synonym to “star-shaped”. The part-of-speech tagger in fact it should have tagged the term as a noun, which had been used in the plot description. The adjective scores were then incorrectly taken into account, illustrating the importance of accurate tagging for classification results.

**SentiWordNet Scoring**

A closer examination of scores extracted from the SentiWordNet database also reveals potential issues. The term “ludicrous” was seen on a misclassified film review, and contains two possible synsets on WordNet. The table below details their synset glosses and SentiWordNet scores:

| Term | Gloss | SentiWordNet Score (Pos, Neg) |
|---|---|---|
| Ludicrous | (adj) farcical, ludicrous, ridiculous (broadly or extravagantly humorous; resembling farce) "the wild farcical exuberance of a clown"; "ludicrous green hair" | (0.5, 0.125) |
| Ludicrous | (adj) absurd, cockeyed, derisory, idiotic, laughable, ludicrous, nonsensical, preposterous, ridiculous (incongruous;inviting ridicule) "the absurd excuse that the dog ate his homework"; "that's a cockeyed idea"; "ask a nonsensical question and get a nonsensical answer"; "a contribution so small as to be laughable"; "it is ludicrous to call a cottage a mansion"; "a preposterous attempt to turn back the pages of history"; "her conceited assumption of universal interest in her rather dull children was ridiculous" | (0.625, 0) |

Both synsets contain generally positive SentiWordNet scores, however consider this term's use on the extract below:

> *"the action in armageddon are so over the top , nonstop , and too ludicrous for words."*

It can be argued that the second synset term should contain a negative orientation, given its association with synonym terms such as "farcical" and "idiotic". However, it appears SentiWordNet assigned positive scores to this term, on the basis the text from the synset gloss is more likely to be associated with a positive oriented term than a negative one. Recalling from section 4.2.5, SentiWordNet scores are expanded to all WordNet terms by applying a classification algorithm based on terms extracted from synset glosses, therefore terms such as "exuberance" and "clown" and the somewhat

ambiguous “laughable” could be influencing the construction method in assigning incorrect scores. The dependence of SentiWordNet scores on term glosses could be a limiting factor in the accuracy of term scores, and the overall classification accuracy of the experiment.

## *7.4. Conclusion*

This chapter presented and discussed the results of the sentiment classification experiment using SentiWordNet, described on Chapter 6. Initially, a baseline classifier method was implemented using a bag-of-word features similar to the one described on (Pang et al, 2002). The SentiWordNet classifier was built using features described on Chapter 5, and an iterative process tested various combinations of feature generation parameters. Next, outlier detection and removal and feature selection was applied to the SentiWordNet classification. Finally a discussion on results obtained and examination of misclassifications was presented. Results and findings were presented on the three key proposed metrics for assessing the experiment: classification accuracy, training set size and training and execution times.

For classification accuracy, the SentiWordNet classifier reported best results of average accuracy of **69.10%** using 3-fold cross-validation, in contrast to a baseline result of 83.90%. The SentiWordNet results are close to the ones reported in (Pang et al, 2002), for a classifier based on simple document statistics and a word list for positive and negative terms manually built for the data set. SentiWordNet results however are built upon a lexicon generated by a semi-automatic method, not dependant on a specific domain, this it could be speculated that the SentiWordNet approach is potentially more generic, more automated and less domain-dependant than manually built word lists.

The parameters affecting SentiWordNet features were also individually tested during the experiment and their effects to overall classification were presented. The best combination found during the experiment calculated SentiWordNet scores according to a linear increasing function of term position in the document, suggesting that terms placed towards the end of a document are more likely to represent author opinion. Intuitively this would translate to an author’s concluding remarks about the film.

During the discussion on SentiWordNet features in Chapter 5, the idea of a threshold value that would triage which term scores were to be used was introduced, in order to address the problem of polysemous terms with several meanings and potentially contradictory SentiWordNet polarity scores. A similar concern is seen in (Dave et al, 2003) on the use of similar term-based lexical resources for opinion mining. In this experiment a threshold value was used in the cases where multiple SentiWordNet scores are found. If the positive and negative scores differed by more than a specified threshold, then it could be argued their polarity is less likely to be ambiguous. The experiment was executed against several threshold values and no benefit was found in discarding terms according to score difference. In other words, a threshold of zero yielded the best results during the experiment, and this approach for distinguishing potentially ambiguous terms did not have a positive effect on classification accuracy. The problem of ambiguous terms however still needs to be addressed and deserves further investigation, and other approaches could have better effect on the SentiWordNet model.

A negation detection algorithm was also implemented to adjust SentiWordNet scores accordingly for negated terms. The algorithm applied is based on the *NegEx* method (Chapman et al, 2001) and yielded a minor improvement to accuracy. The effects of negation detection have been extensively studied on the area of medical records (Chapman et al, 2001; Huang et al, 2007; Mutalink et al, 2001), and further explorations on other domains such as film reviews could yield better results for SentiWordNet scores.

Finally, the best choice of document segmentation found was to divide the document equally into 10 segments and calculate scores for each segment individually. Similar to the linear scoring function detailed above, this approach attempts to detect and highlight parts of the document with stronger opinion.

The final step of the experiment was to refine results obtained by SentiWordNet by performing outlier removal and feature selection. These steps improved the results for 3-fold cross validation accuracy to the final result of **69.10%**. The feature removal method has also illustrated a weak correlation (using chi-squared test) between

features that count negated terms and negative scores for adverbs to the document's opinion polarity. This finding may suggest negation expressions as a stylistic resource are equally used on both positive and negative biased reviews, whereas it has been noted that other features related to adverbs and verbs do not contain enough scores on the SentiWordNet database and therefore add little information to the model.

For different sizes of training set, the SentiWordNet approach did not yield substantial differences in accuracy, and results remained within or below 69.10%. The 100-fold cross-validation experiment however has shown similar results with and without outlier detection and removal from the training process, suggesting there is less need for this step as training data becomes more available. Alternatively, the outlier detection step could be tuned for larger training data sets accordingly.

Execution time for the SentiWordNet approach remained substantially smaller than the baseline classifier, due to the much smaller dimensionality of the feature set built from SentiWordNet, in comparison to a word vector based on unigrams, suggesting this approach may be better suited for cases where making a prediction as timely as possible is of paramount importance. Training times however did not differ considerably on the baseline and SentiWordNet cases, in line with the expected theoretical behaviour of support vector machines, and indeed the baseline method outperformed SentiWordNet on training times on nearly all cases.

A review of documents that were incorrectly classified by the SentiWordNet classifier revealed several challenges and opportunities for improvement of this method. The effect of order of opinion related terms in a sentence is important, as is the case with the phenomenon of thwarted expectations, where the author builds up expectations on positive or negative aspects of the film, only to be frustrated by an opinion with reverse polarity. This effect suggests scoring term polarity alone with no regard to their placement in the narrative can be misleading to assess overall document sentiment. The use of colloquial expressions from spoken English, acronyms and word shortenings not present in SentiWordNet also influence scores, and pre-processing techniques might be able to help in detecting such cases. In addition, the accuracy of part of speech tagging was noted to influence overall accuracy, and some scores for SentiWordNet terms were found to be misleading, suggesting the automated method

for building the SentiWordNet database has an over reliance on WordNet glosses, and can be improved for the purposes of detecting term polarity.

The results obtained from the SentiWordNet experiment highlighted a number of challenges and opportunities for improving the method, the next chapter concludes this dissertation by summarising the findings from the literature review, experiment design and execution, and discuss future research work possibilities.

# 8. CONCLUSION

This chapter concludes this dissertation's research. The introductory section reviews the motivation, potential benefits and key challenges of opinion mining and sentiment classification, followed by a discussion on the research objectives and achievements. Results on the experiment using SentiWordNet for sentiment classification are reviewed, with concluding remarks on the obtained results. Opportunities for future research work are presented, and the chapter concludes with final remarks on the work performed.

## *8.1. Introduction*

This dissertation is a research in the field of opinion mining that evaluates the application of the SentiWordNet lexical resource for sentiment classification in documents. Driven by the increasing availability of subjective information in digital format on resources publicly available on the internet and in corporate information systems, the field of opinion mining entails the use of automated methods for detecting subjective content within text resources, and has applications in a variety of fields from online advertising systems to search engines and market research. In many instances, sentiment information forms a key component in building the knowledge required for effective decision making, and is therefore a subject of concern to the field of knowledge management.

To achieve the aims of opinion mining, and to effectively perform its task, the field draws from resources on knowledge discovery, data mining and text mining. In addition, because of the complexity and nuances of subjective language, opinion mining is a rich field for the application of natural language processing techniques. To further understand the challenges of opinion mining, a review of the state of the art of research in the above fields was conducted as part of this dissertation's research, and so was its relationship to the objectives of the broader field of knowledge management.

The experiment performed as part of this research produced not only results that can be used for future reference on the field, but also revealed several challenges where future research might be of interest. In the next sections, those are presented in more details.

## *8.2. Research Overview and Objectives*

The research performed as part of this project aimed at reviewing the state of the art in the areas of knowledge management, knowledge discovery and opinion mining. This review was then employed to the design and implementation of an experiment aiming at assessing the effectiveness of the SentiWordNet lexical resource for the purposes of sentiment classification. During this research, the following objectives were achieved.

- Review of the literature in the area of knowledge management, in particular exploring the importance of knowledge creation and knowledge discovery to the success of modern organisations.

- Review of research in the fields of knowledge discovery and data mining, techniques, challenges and applications to knowledge discovery.

- Review of research literature on text mining and opinion mining, exploring applications of opinion mining to computer systems, and their relationships to the goals of knowledge management. Exploration of opinion mining techniques for detecting sentiment orientation in documents, and the use of lexical resources for opinion mining.

- Evaluation of the SentiWordNet lexical resource, and design of a model that extracts features from text documents using the SentiWordNet for the purposes of sentiment classification.

- Design and execution of a sentiment classification experiment that tests the effectiveness of SentiWordNet according to the criteria of classification accuracy, runtime and sensitiveness to training data, and compares results to a well known baseline classifier documented in the literature.

- Comparisons of results obtained and exploration of challenges and limitations of the approach proposed by the experiment.

## *8.3. Experiment Results*

As outlined on Chapter 6, the results for the sentiment classification experiment were measured according to three key factors: classification accuracy, training set sizes and timings for training and execution. For classification accuracy, the best results obtained by the SentiWordNet classifier using 3-fold cross validation were **69.10%**. This compares to the 83.90% obtained using the baseline classifier based on bag-of-word features. The SentiWordNet results are similar to the classifier based on simple document statistics and a list for positive and negative words presented in (Pang et al, 2002). The table below illustrates how SentiWordNet compares to other published results in the area tested with the same polarity data set used by this experiment. Where possible, the results reported reflect an experiment using 3-fold cross validation.

| **Method** | **Accuracy** | **Source** |
|---|---|---|
| Support Vector Machines and *Bigrams* word vector | 77.10% | (Pang et al, 2002) |
| Naïve Bayes + Parts of Speech | 77.50% | (Salvetti et al, 2004) |
| Word vector with Adjectives Only | 77.70% | (Pang et al, 2002) |
| Support Vector Machines and *Unigrams* word vector | 82.90% | (Pang et al, 2002) |
| Unigrams + Subjectivity Detection | 87.15% | (Pang et al, 2004) |
| | | |
| Positive/Negative Word Lists | 69.00% | (Pang et al, 2002) |
| Term counting from General Enquirer Dictionary + Linguistic Features | 67.80% | (Kennedy et al, 2006) |
| **SentiWordNet** | **69.10%** | |

**Table 40 - Accuracy Comparison with Published Research**

Clearly, results obtained using the SentiWordNet approach remain below the state-of-the-art techniques, but within close range to other similar approaches. At first sight this may suggest an upper bound in sentiment classification techniques that rely solely on term polarity information. Indeed, further investigation on misclassified reviews

revealed some limitations of this approach, but it also revealed opportunities for improvement of this method. Initially, it was noticed that some of the SentiWordNet term scores did not carry the expected polarity. This can be attributed to the method by which SentiWordNet is built, which relies on WordNet glosses for making a classification prediction, and can lead to inaccuracies.

It was also noticed that the current method used for extracting information was taking into account irrelevant terms such as films and actor names commonly referenced on text but of little relevant to detect author opinion. In addition, further refinements in capturing English expressions and acronyms would assist in detecting opinion on the more colloquially written reviews.

The issue of word sense disambiguation was not addressed as part of this experiment, and improvements in this area can certainly assist in making the right decision for SentiWordNet score where terms have more than one meaning. Finally, the issue of *thwarted expectations* – where a positive expectation is built up using words with positive orientation, only to be frustrated by a negative opinion - was present on many examples, and impairs the correct prediction of reviews.

The results obtained for training data set size revealed similar patterns for increasing numbers of training data, thus indicating the need for increasing the predictive power of the model, possibly by adding more features and improving the quality of existing features.

There was no substantial gain on training times when comparing SentiWordNet with the baseline classifier, however the execution time required for a prediction on a new document to be made was much faster on SentiWordNet. This can be attributed to the considerably lower number of features used in the SentiWordNet model, which makes this model a good candidate for systems where near real time predictions are necessary.

As previously discussed on Chapter 3, knowledge discovery is an iterative process, where findings may lead to new insights and different lines of investigation. The findings obtained from the SentiWordNet experiment highlighted a number of

challenges and potential improvements in data preparation, data mining and model improvement. The next sections outline key findings and additions to the body of knowledge in opinion mining research, and propose further research in some of the above aspects, along with alternative approaches for applying SentiWordNet information to opinion mining.

## *8.4. Additions to the Body of Knowledge*

As outcomes of this dissertation's research and experiment results, the following findings can be highlighted as contributions to the body of knowledge in the area of opinion mining.

The key part of this project's experiment was to perform sentiment classification using features built from the SentiWordNet database of term polarity scores. This project presents a proposed design for a set of features, and classification results obtained by experimentation using the polarity data set. These results can be used as reference for future research in the area on this class of sentiment classification technique.

The experiment also has shown that weighting the polarity score of terms as a function of its location in the document has an effect on overall accuracy, as final remarks on a review tend to carry heavier sentiment content, further validating comments from (Pang et al, 2008; Pang et al, 2002) that document structure is a meaningful aspect for measuring a document's sentiment orientation.

The research has also demonstrated the connection with negation expressions in text, and the accuracy of final results when performing sentiment classification with SentiWordNet, with minor improvements obtained with a simple negation detection algorithm based on the work of (Chapman et al, 2001), suggesting further improvements in this area may lead to better results in sentiment classification.

One potential limiting factor on the SentiWordNet approach unearthed during this research was the reliance on WordNet *glosses,* used by SentiWordNet for automatically building term polarity scores. This dependency could lead to

inaccuracies in term polarity scores depending on the gloss's contents, and could affect the overall accuracy of sentiment classification when using this resource.

## *8.5. Future Work & Research*

The research performed on knowledge management, data mining, opinion mining and SentiWordNet, along with the outcomes obtained through experimentation and the analysis of results have uncovered opportunities for future research which could lead to interesting results, and are explored in more details in this section.

### 8.5.1. SentiWordNet Features

The SentiWordNet set of features proposed in Chapter 5 and used during this dissertation's experiment can be further refined by introducing word sense disambiguation in conjunction with SentiWordNet for more accurately scoring terms found in text. As discussed in the experiment results, the approach proposed in this research did not lead to any gains in accuracy, suggesting there is further room for improvement in this topic.

Also, incorporating the detection of colloquial expressions and detection of entities such as actors, locations and film names are relatively straightforward approaches that may improve results further by removing potentially noisy scores from the data set.

Another important aspect found during experimentation was the inclusion of predominantly descriptive sections of text within the final SentiWordNet scores, which may not carry strong opinion content but could lead to inaccuracies in the final classification results. Including detection of subjective text as a pre-processing step ahead of building the SentiWordNet data set may further improve results using this approach.

It was also found during the experiment that inaccuracies stemming from incorrectly tagging parts of speech could lead to poor classification results, since scoring terms using SentiWordNet relies heavily on this information. Experimenting with different part-of-speech tagging techniques could also lead to improved results.

The negation algorithm implemented as part of the experiment is a simplified version of the *NegEx* method presented on (Chapman et al, 2001), and further improvements that enable the detection of more subtle negating and opinion changing expressions in text could lead to improved results.

### 8.5.2. Classification Results

It has been observed in (Read, 2005; Pang et al, 2008) that training machine learning techniques for sentiment classification tend to be specific to a domain such as film reviews or editorials, and to specific topics like films or digital cameras. These are not immediately applicable across other domains or topics, or when applied, generate poorer classification results. One aspect of applying SentiWordNet is that it is based on WordNet synsets, and likely to be more generic and more applicable to other domains, and could serve as a baseline classifier when no domain or topic specific classifier exists. It would be interesting to assess the effects of classification performance where the training and validation data sets belong to different domains, since development on this area may lead to more widely applicable classifiers.

By the same principle, it could be argued that whereas classification accuracies obtained with SentiWordNet are inferior to those of other techniques, it may contain important information for the detection of document sentiment that is being missed on other methods. Thus, *combining* the results of multiple classifiers might lead to improvements in overall performance and better methods. This is the subject of research in *multiple classifier systems* (MCS), with some techniques surveyed in (Kittler et al, 1998; Provost et al, 2001) and examples of such technique implemented in the opinion mining domain reported in (Mullen et al, 2004) and (Kennedy et al, 2006).

Another interesting aspect of SentiWordNet is its application to the area of *bootstrapping* classifiers when little training data is available. SentiWordNet could be applied as a high-precision domain-independent classifier used as the initial stage for the creation of larger training sets for sentiment classification.

Finally, the results of this experiment were obtained by carrying out a limited search on parameter combinations, and no extensive classifier tuning was performed. Potentially better results could be obtained with more extensive investigation of possible parameter combinations either by exhaustive searching – an approach potentially prohibitive in time and resources – or by applying more sophisticated parameter searching approaches, as described in (Hand et al, 2001).

### 8.5.3. Knowledge Management Research

Another topic that deserves further exploration relates to the less technical aspects of opinion mining, and their application in knowledge discovery, and suitability to knowledge management initiatives. For instance, due to widespread availability of text information in digital format, text mining applications have been surveyed in the literature and closely linked to knowledge management objectives (Marwick et al, 2001; Feldman et al, 1998). Assessing the usage of an opinion mining component as a requirement on such applications would be beneficial not only to further encourage opinion mining research, but also to bring additional functionality to knowledge management systems.

## *8.6. Conclusions*

We conclude from the results obtained by the experiment that whereas results in accuracy for the SentiWordNet approach were below current state-of-the-art methods, it also highlighted aspects where further research can generate potential improvements. This, coupled with the lower dimensionality of the SentiWordNet data set, and its relatively lower dependence on domain information could lead to more attractive models for real world applications.

Knowledge discovery methodologies will no doubt play an important part in such developments, as attention to all aspects of discovery are needed for effective results, and the iterative nature of knowledge discovery will render future opportunities for development as more results become available. Likewise, the development of data mining algorithms and advances in natural language processing also have a part to play in the improvement of opinion mining techniques, as this is a research area that combines efforts from all the above fields.

In the context of knowledge management, it is clear that opinion information adds a new dimension to what can be extracted from textual data, and has the potential of improving an organisation's ability to create new knowledge through knowledge discovery approaches. As reviewed on Chapter 2, these are crucial aspects for effective decision making processes and for ensuring companies remain creative and competitive.

### 8.6.1. Final Remarks

The field of opinion mining is an exciting new area of research with the potential for a number of real world applications where discovering opinion information is relevant to making better decisions. The development of techniques for document sentiment classification is one important component of this area and further developments will certainly impact the quality and speed at which knowledge derived from opinion information can be created, with implications to companies' ability to compete and respond to customer demands.

The research presented in this dissertation has assessed the viability of performing document sentiment classification by using SentiWordNet as an automatically built lexical resource of opinion polarity in terms. The research also highlighted further developments in this field are possible, giving it the potential for being an attractive approach for a number of real world applications.

# REFERENCES


Alavi M, Leidner D. E, (2001) “Knowledge Management and Knowledge Management Systems: Conceptual Foundations and Research Issues”, *MIS Quarterly*, Vol. 25 No. 1, 107-136, March 2001.

Alavi M, Leidner D.E. (1999) "Knowledge management systems: issues, challenges, and benefits", *Communications of the AIS*, 1999.

Ali, K. M. Pazzani, M. J. (1993) “HYDRA: A Noise-tolerant Relational Concept Learning Algorithm”, *International Joint Conference in Artificial Intelligence*, Vol. 13, Number 2, 1993.

Allix N. (2003) “Epistemology and Knowledge Management Concepts and Practices”, *Journal of Knowledge Management Practice*, April 2003.

Alpoydin E. (2004) “Introduction to Machine Learning”, *MIT Press*, 2004.

Apte C., Damerau, F. J., Weiss, S. M. (1994) “Automated learning of decision rules for text categorization”, *ACM Transactions on Information Systems*, No. 12, Vol. 3, 233–251, 1994.

Arthur B. (1996) “Increasing Returns and the New World of Business”, *Harvard Business Review, 101-109, July-August 1996.*

Awad E, Ghaziri H, (2004) “Knowledge Management”, *Pearson – Prentice Hall*, 2004.

Boiy E, Hens P, Deschacht K, Moens M, (2007) "Automatic Sentiment Analysis in On-line Text", *Proceedings ELPUB 2007 Conference in Electronic Publishing - Vienna*, Austria - June 2007.

Boser B, Guyon I, Vapnik V. (1992) “A Training Algorithm for Optimal Margin Classifiers”, *Proceedings of the 5th Annual ACM Workshop on Computational Learning Theory*, pp. 144-152, March, 1992.

Brachman R, Khabaza T, Kloesgen W, Piatetsky-Shapiro G, Simoudis E, (1996) “Mining Business Databases”, *Communications of the ACM*, Vol. 39, No 11, 42-48, November, 1996.

Brants T. (2000) "TnT--a statistical part-of-speech tagger", *Proceedings of the sixth conference on Applied natural language processing*, 224-231, 2000.

Brijs T, Swinnen G, Vanhoof K, Wets G. (1999) “Using Association Rules for Product Placement Decisions: A Case Study”, *Proceedings of the fifth ACM SIGKDD International Conference*, 1999.

Burges C. (1998) “A Tutorial on Support Vector Machines for Pattern Recognitrion”, *Data Mining and Knowledge Discovery*, Vol. 2, pp. 121-167, 1998.

Catmull E. (2008) “How Pixar Fosters Collective Creativity”, *Harvard Business Review*, 65-72, September 2008.

Chapman P, Clinton J, Kerber R et al, (2000) “CRISP-DM 1.0 – Step by Step Data Mining Guide”, *The CRISP Consortium*, from http://www.crisp-dm.org/Process/index.htm.

Chapman W, Bridewell W, Hanbury P, Cooper G, Buchanan B. (2001) “Evaluation of Negation Phrases in Narrative Clinical Report”, *Proceedings of 2001 AMIA Symposium*, pp 105-109, 2001.

Clark P, Niblett T. (1989) “The CN2 Induction Algorithm”, *Machine Learning*, Springer, 1989.

Cody W, Kreulen J. T, Krishna V, Spangler S W, (2002) “The Integration of Business Intelligence and Knowledge Management”, *IBM Systems Journal*, Vol. 41, No. 4, pp. 697-713, 2002.

Cole R, (1998) “Introduction”, *California Management Review*, Vol. 40, No. 3, 15-21, Spring 1998.

Conrad J, Al-Kofahi K. (2005) “Effective document clustering for large heterogeneous law firm collections”, *Proceedings of the 10th international conference on Artificial intelligence and law*, pp. 177-187, 2005.

Corney M. de Vel, O., Anderson A, Mohay G. (2002) "Gender-preferential text mining of e-mail discourse", *18th Annual Computer Security Applications Conference*, 2002.

Csomai A, Rosenzweig J, Mihalcea R. (2007) “WordNet Bibliography”, WordNet Portal, accessed online Jan/2009 from [http://lit.csci.unt.edu/~wordnet/].

Cui H, Mittal V, Datar M. (2006) “Comparative Experiments on Sentiment Classification for Online Product Reviews”, *Proceedings of the National Conference on Artificial Intelligence*, AAAI Press, Vol. 21; pp. 1265-1270, 2006.

Dave K, Lawrence S, Pennock D. (2003) “Mining the Peanut Gallery: Opinion Extraction and Semantic Classification in Product Reviews”, *Proceedings of the 12th International conference on the World Wide Web - ACM WWW2003*, May 20-24, Budapest, Hungary, 2003.

Davenport T, Prusak L, (1998) “Working Knowledge – How Organisations Manage What They Know”, *Harvard Business School Press*, 1998.

Devitt A, Ahmad K, (2007) “Sentiment Polarity Identification in Financial News: A Cohesion Based Approach”. *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics, pages 984–991*,Prague, Czech Republic, June 2007.

Don A, Zheleva E, Gregory M, Tarkan S, Auvil L, Clement T, Shneiderman B, Plaisant, C. (2007) “Discovering interesting usage patterns in text collections: integrating text mining with visualization”, *University of Maryland Human-Computer*

*Interaction Labs, Technical report 2007-08,* May 2007. [http://www.cs.umd.edu/hcil/bioinfovis/]

Dorre J, Gerstl P, Seiffert R. (1999) “Text mining: finding nuggets in mountains of textual data”, *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining,* 1999.

Drucker H, Wu D, Vapnik V. (1999) “Support Vector Machines for SPAM Categorization”, *IEEE Transactions on Neural Networks,* Vol. 10, No. 5, pp 1048-1054, September 1999.

Drucker P. (1995) “The Information Executives Truly Need”, *Harvard Business Review,* January-February 1995, pp. 55-62, 1995.

Esuli A, Sebastiani F, (2006) "SentiWordNet: A Publicly Available Lexical Resource for Opinion Mining", *Proceedings from International Conference on Language Resources and Evaluation (LREC),* Genoa 2006.

Fahey L, Prusak L, (1998) “The Eleven Deadliest Sins of Knowledge Management”, *California Management Review,* Vol. 40, No 3, pp. 265-276, Spring 1998.

Farhoomand A, Drudy D H, (2002) “Managerial Information Overload”, *Communications of the ACM,* Vol. 45, No. 10, pp. 127-131, October 2002.

Fayyad, U, Haussler, D, and Stolorz, P. (1996) “Mining scientific data.” *Communications of the ACM,* No. 39, Vol. 1, November 1996, 51-57.

Fayyad, U, Piatetsky-Shapiro, G, Smyth, P. (1996) "The KDD process for extracting useful knowledge from volumes of data". *Communications of the. ACM,* No. 39, Vol. 11, November 1996, 27-34.

Fayyad, U. M., Weir, N., and Djorgovski, S. (1993) “Automated cataloguing and analysis of sky survey image databases: the SKICAT system.” *Proceedings of the*

*Second international Conference on information and Knowledge Management (CIKM '93)* Washington, D.C., United States, November 01 - 05, 1993.

Fayyad, U. Piatetsky-Shapiro, G. Smyth, P, (1996) "From Data Mining to Knowledge Discovery in Databases". *AI Magazine*, Vol. 17; No. 3, (1996) pages 37-54.

Feldman R, Dagan, I. (1995) "Knowledge discovery in textual databases (KDT)", *Proceedings of the First International Conference on Knowledge Discovery and Data Mining (KDD-95)*, pp. 112-117, 1995.

Feldman R, Fresko M, Hirsh H, Aumann Y, Liphstat O, Schler Y, Rajman M. (1998) "Knowledge Management: A Text Mining Approach", *Proceedings of the 2nd international conference on Practical Aspects of Knowledge Management – PAKM'98*, Basel, Switzerland, Oct. 1998.

Feldman R, Dagan I, Hirsh I. (1998-b), "Mining Text Using Keyword Distributions", *Journal of Intelligent Information Systems*, Vol. 10, pp. 281-300, 1998.

Fellbaum C, Gross D, Miller K. J, (1990) "Adjectives in WordNet" *International Journal of Lexicography*, vol. 3, no. 4, pp. 235-244, January 1990.

Forman G, Kirshenbaum E, Suermondt J, (2006) "Pragmatic Text Mining: Minimizing Human Effort to Quantify Many Issues in Call Logs", *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '06-Philadelphia*), pp. 852-861, August 2006.

Forman G. (2002), "An extensive empirical study of feature selection metrics for text classification", *The Journal of Machine Learning Research*, Vol. 3, pp. 1289-1305, 2003.

Fry B, (2008) "Visualising Data", *O'Reilly Press*, 2008.

Gamon, M., (2004) "Sentiment classification on customer feedback data: noisy data, large feature vectors, and the role of linguistic analysis." *In Proceedings of the 20th*

*international conference on Computational Linguistics*. Geneva, Switzerland: Association for Computational Linguistics, p. 841, 2004.

Gargano M L, Raggad B G, (1999) "Data Mining – A Powerful Information Creating Tool", *OCLC Systems and Services, Vol 15, No 2, 81-90, MBC University Press,* 1999.

Garside, R**.** (1987). "The CLAWS Word-tagging System.", *The Computational Analysis of English: A Corpus-based Approach*. London: Longman.

Ghani R, Probst K, Liu Y, Krema M, Fano, A. (2006) "Text mining for product attribute extraction", *ACM SIGKDD Explorations Newsletter*, Vol. 8, No. 1, pp. 41-48, 2006.

Goebel M, Gruenwald L. (1999) "A Survey of Data Mining and Knowledge Discovery Software Tools", *ACM SIGKDD Explorations*, pp. 20-33, Vol. 1, Issue 1, June, 1999.

Hahn J, Subramani M. (2000) "A Framework of Knowledge Management Systems: Issues and Challenges for Theory and Practice", *Proceedings of the Twenty-first International Conference on Information Systems,* Brisbane, Australia 2000.

Hand D, Mannila H, Smyth P, 2001, "Principles of Data Mining", *The MIT Press*, Cambridge Massachusetts, 2001.

Hansen M, Nohria N, Tierney T, (1999) "What's Your Strategy for Managing Knowledge?" , *Harvard Business Review, March*-April, 106-116, 1999.

Hayes P, Weinstein S.P. (1990) "Construe-TIS: A system for Content-based indexing of databas news stories", *Proceedings of the conference in Innovative Applications of Artificial Intelligence - IAAI90,* 1990.

Hearst, M. A. (1999) "Untangling text data mining." *In Proceedings of the 37th Annual Meeting of the Association For Computational Linguistics on Computational Linguistics*, College Park, Maryland, June 20 - 26, 1999.

Herschel R, Jones N E, (2005) “Knowledge Management and Business Intelligence: The Importance of Integration”, *Journal of Knowledge Management*, Vol. 9, No. 4, 45-55, 2005.

Hipp J, Guntzer U, Nackaeizadeh G. (2000) “Algorithms for Association Rule Mining: A General Survey and Comparison”, *SIGKDD Explorations*, Vol. 2, Issue 1, 58-64, July 2000.

Ho Y, Agrawala A. (1968) “On Pattern Classification Algorithms – Introduction and Survey”, *Proceedings of the IEEE*, Vol. 56, No. 12, December 1968.

Hochheiser, H, Baehrecke, E.H, Mount, S.M, Shneiderman, B. (2003) “Dynamic Querying for Pattern Identification in Microarray and Genomic Data” *Proceedings 2003 IEEE International Conference on Multimedia and Exposition*, 2003.

Hoffman, P., Grinstein, G., Marx, K., Grosse, I., and Stanley, E. 1997. “DNA visual and analytic data mining”. *Proceedings of the 8th Conference on Visualization '97 (Phoenix, Arizona, United States, October 18 - 24, 1997).*

Hofmann M, (2003) “The Development of a Generic Data Mining Life Cycle (DMLC)”, *M.Sc Dissertation*, Dublin Insitute of Technology, 2003.

Holsapple C, (2002) “Knowledge and Its Attributes”, *Handbook on Knowledge Management*, 165-188, Springer, 2002.

Holsapple C, Joshi K, (2002) “Knowledge Manipulation Activities: Results of a Delphi Study”, *Information and Management*, No 39, 477-490, Elsevier Science, 2002.

Holsapple C.W, Joshi K.D, (1999) “Description and Analysis of Existing Knowledge Management Frameworks”, *Proceedings of the 32nd Hawaii International Conference on System Sciences, IEEE, 1999.*

Horrigan J. (2008) “Online Shopping”, *Pew Internet and American Life Project – Research Report*, February, 2008.

Hotho A, Nurnberger A, Paaß G. (2005) “A Brief Survey of Text Mining”, *LDV Forum-GLDV Journal for Computational Linguistics and Language Technology*, Vol. 20, No. 1, pp.19.62, 2005.

Huang Y, Lowe H. (2007) “A Novel Hybrid Approach to Automated Negation Detection in Clinical Radiology Reports”, *Journal of the American Medical Informatics Association*, Vol. 14, No. 3, May/June 2007.

Ide N., Véronis J. (1998) “Introduction to the special issue on word sense disambiguation: the state of the art”. *Computational Linguistics.* No. 24, Vol. 1, March 1998, pp. 2-40.

Jin, X., Li, Y., Mah, T., and Tong, J. (2007) “Sensitive webpage classification for content advertising.” *Proceedings of the 1st international Workshop on Data Mining and Audience intelligence For Advertising ADKDD '07,* San Jose, California, ACM, New York, NY, pp. 28-33, August 2007.

Joachims, T. (1998) “Text Categorization with Support Vector Machines: Learning with Many Relevant Features*”*. *Proceedings of the European Conference on Machine Learning (ECML), Springer, 1998*.

Joachims, T. (1996) "A probabilistic analysis of the Rocchio algorithm with TFIDF for text categorization.", *University of Pittsburg Carnegie-Mellon Dept. of Computer Science*, Research Report, 1996.

John G.H, Miller P. (1996) “Building Long/Short Portfolios Using Rule Induction”, *Proceedings of the IEEE International Conference on Computational Intelligence for Financial Engineering*, New York City, March, 1996.

Kankanhalli A, Tanudidjaja F, Sutanto J, Tan B, (2003) “The Role of IT in Successful Knowledge Management Initiatives”, *Communications of the ACM,* September 2003, Vol. 46, No. 9, 69-73, 2003.

Kao, A., Quach, L., Poteet, S., Woods, S. (2003) "User assisted text classification and knowledge management", *Proceedings of the twelfth international conference on Information and knowledge management*, 524-527, 2003.

Kolcz J, Alspector E., (2001) "SVM-based filtering of e-mail spam with content-specific misclassification costs", *Proceedings of the Workshop on Text Mining (TextDM'2001)*, 2001.

KDNUGGETS (2006) "Poll: Data Mining Software", *KDNuggets Portal*, Accessed Feb/2009: [http://www.kdnuggets.com/polls/2006/data_mining_analytic_tools.htm], 2006.

KDNUGGETS (2007) "Poll: Data Mining Software", *KDNuggets Portal*, Accessed Feb/2009 : [http://www.kdnuggets.com/polls/2007/data_mining_software_tools.htm] , 2007.

KDNUGGETS (2008) "Poll: Data Mining Software", *KDNuggets Portal*, Accessed Feb/2009 from [http://www.kdnuggets.com/polls/2008/data-mining-software-tools-used.htm], 2008.

KDNUGGETS (2009) "Software for Data Mining, Analytics and Software Discovery", *KDNuggets Portal*, Accessed online on Feb/2009 [http://www.kdnuggets.com/software/index.html], 2009

Keim D, Schneiderwind J. (2007) "Introduction to the Special Issue in Visual Analytics", SIGKDD Explorations, Vol. 9, Issue 2, 2007.

Kim S, Hovy E. (2004) "Determining the Sentiment of Opinions." *In Proceedings of Conference on Computational Linguistics (COLING-04).* pp. 1367-1373. Geneva, Switzerland, 2004.

Kittler J, Hatef M, Duin R, Matas J. (1998) "On Combining Classifiers", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 20, no. 3, pp. 226-239, March 1998.

Knox E.M, Ng R.T. (1998) “Algorithms for mining distance-based outliers in large datasets”, Proceedings of the International Conference on Very Large Databases, Vol. 1, pp. 392-403, 1998.

Kolyshkina I, Simoff S. (2007) “Customer Analytics Projects: Addressing Existing Problems with a Process that Leads to Success”, *Proc. 6th Australian Data Mining Conference (AusDM’07)*, Australia, 2007.

König, A. C. and Brill, E. (2006). “Reducing the human overhead in text categorization.” *Proceedings of the 12th ACM SIGKDD international Conference on Knowledge Discovery and Data Mining - KDD '06*, Philadelphia, PA, USA, ACM, New York, NY, pp. 598-603, August, 2006.

Kroeze, J. H., Matthee, M. C., and Bothma, T. J. (2003) “Differentiating data- and text-mining terminology”. *Proceedings of the 2003 Annual Research Conference of the South African institute of Computer Scientists and information Technologists on Enablement Through Technology*, September 17 - 19, 2003.

Kulkarni S, Lugosi G, Venkatesh S. (1998) “Learning Pattern Classification – A Survey”, *IEEE Transactions on Information Theory*, Vol. 44, No. 6, October 1998.

Lent B, Agrawal R, Srikant R. (1997) “Discovering trends in text databases”, *Proc. of the 3rd Int'l Conference on Knowledge Discovery in Databases and Data Mining*, Newport Beach, California, 1997.

Lim T, Lo W, Shih Y. (2000) “A Comparison of Prediction Accuracy, Complexity and Training Time of Thirty-Three Old and New Classification Algorithms”, *Machine Learning*, No. 40, 203-229, 2000.

Liu B, Chang K.C. (2004) “Editorial: Special Issue on Web Content Mining”. *ACM SIGKDD Explorations Newsletter*, Vol. 6, No. 1, pp. 1-4, 2004.

Liu H, Yu L, (2005) “Towards Integrating Feature Selection Algorithms for Classification and Clustering”, *IEEE Transactions on Knowledge and Data Engineering*, Vol 17, No 4, 491-502 April 2005.

Livingston, G, Li X, Li G, Hao L, Zhou J. (2003) “Using Rule Induction Methods to Analyze Gene Expression Data” , *Proceedings of the 2003 IEEE Bioinformatics Conference*, August, 2003.

Loper E, Bird S. (2002) "Nltk: The natural language toolkit", *Proceedings of the ACL Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics*, 62-69, 2002.

Ma Y, Liu B, Wong C.K. (2000) “Web for Data Mining: Organizing and Interpreting the Discovered Rules Using the Web”, *SIGKDD Explorations*, Vol. 2, Issue 1, 16-23, July 2000.

MAKE. (2007) “2007 Global Most Admired Knowledge Enterprises (MAKE) Report – Executive Summary”, *The KNOW Network, Accessed October 5th 2008*, Accessed Oct/2008 from [http://www.knowledgebusiness.com].

Marcus, M. P. Marcinkiewicz, M. A. and Santorini, B. (1993). “Building a large annotated corpus of English: the penn treebank.” *Computation Linguistics*. No. 19, Vol. 2 (June 1993), 313-330.

Marwick A D. (2001) “Knowledge Management Technology”, *IBM Systems Journal*, Vol 40, No 4, pp. 814-830, 2001.

McCallum, A., Nigam, K. (1998) "A comparison of event models for naive bayes text classification", *AAAI-98 workshop on learning for text categorization*, Vol. 7, March 1998.

McDermott R. (1999) “Why Information Technology Inspired but Cannot Deliver Knowledge Management”, *California Management Review*, Vol. 41, No. 4, pp. 103-117, Summer 1999.

McDonald, D.W, Ackerman M.S. (1998) "Just talk to me: a field study of expertise location", *Proceedings of the 1998 ACM conference on Computer supported cooperative work*, pp. 315-324, 1998.

McKnight W. (2005) “Text Data Mining in Business Intelligence”, *Information Management Magazine*, January 1st, 2005. Accessed February 2009 from [http://www.information-management.com/issues/20050101/1016487-1.html].

Meyer, T.A. and Whateley, B. (2004) " SpamBayes: Effective open-source, Bayesian based, email classification system", *Proceedings of the First Conference on Email and Anti-Spam (CEAS)*, 2004.

Mierswa I, Wurst M, Klinkenberg R, Scholz M, Euler T, (2006) “YALE: Rapid Prototyping for Complex Data Mining Tasks”, *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-06)*, 2006.

Mihalcea R, Strapparava C. (2005) “Making Computers Laugh: Investigations in Automatic Humour Recognition”, *Joint Conference on Human Language Technology/Empirical Methods in Natural Language Processing (HLT/EMNLP)*, 2005.

Miller G. A., Beckwith R., Fellbaum C, Gross D, Miller K. J, (1990) "Introduction to Wordnet: An on-line lexical database" *International Journal of Lexicography*, vol. 3, no. 4, pp. 235-244, January 1990.

Mobasher B, Dai H, Luo T,Nakagawa M. (2002) “Discovery and evaluation of aggregate usage profiles for web personalization”, *Data Mining and Knowledge Discovery*, No. 1, Vol. 6, pp. 61-82, 2002.

Moschitti A, Basili R. (2004) "Complex linguistic features for text classification: A comprehensive study." *Lecture notes in computer science*, Vol. 2997, 181-196, 2004.

Mullen T, Collier N. (2004) “Sentiment Analysis using Support Vector Machines with diverse Information Sources”, *Proceedings of EMNLP*, 2004.

Mutalik P, Deshpande A, Nadkardi P. (2001) “Use of General-Purpose Negation Detection to Augment Concept Indexing of Medical Documents”, *Journal of the American Medical Informatics Association*, Vol. 8, No. 6, November/December 2001.

Nasukawa T, Nagano T. (2001) “Text Analysis and Knowledge Mining System”, *IBM Systems Journal*, Vol. 40, No. 4, 2001.

Nasukawa, T, Yi, J. (2003) “Sentiment analysis: capturing favorability using natural language processing.” *Proceedings of the 2nd international Conference on Knowledge Capture K-CAP '03*, ACM, New York, NY, pp.70-77, 2003.

Nilsson N. (1996) “Introduction to Machine Learning”, *Draft of Proposed Textbook – Dept. of Computer Science*, Stanford University, 1996 [http://robotics.stanford.edu/~nilsson/MLDraftBook/MLBOOK.pdf].

Nirenburg, S. (1997) “Bar Hillel on Machine Translation: Then and Now”, *In Memoriam Yehoshua Bar Hillel M. Caspi and E. Shamir (eds.),* Computing Research Laboratory. New Mexico State University. Las Cruces, USA.

Nonaka I, Konno N, (1998) “The concept of ‘Ba’: building a Foundation for Knowledge Creation”, *California Management Review*, Vol. 40, No. 3, 40-54, Spring 1998.

Nonaka I. (1991) “The Knowledge Creating Company”, *Harvard Business Review*, pp. 96-104, November-December 1991.

Nonaka I. (1994) “A Dynamic Theory of Organisational Knowledge Creation”, *Organisation Science*, 14-37, February 1994.

Nonaka I. (1995) “The Knowledge-creating Company: How Japanese Companies Create the Dynamics of Innovation”, *Oxford University Press, 1995.*

ORACLE (2007) “Oracle Data Mining with 11g: Know More, Do Less, Spend Less”, *Oracle White Paper*, [http://www.oracle.com/technology/products/bi/odm], Accessed February 2009, 2007.

Ordonez C. (2006) “Association Rule Discovery with the Train and Test Approach for Heart Disease Prediction”, *IEEE Transactions on Information Technology in Biomedicine*, Vol. 10, No. 2, April 2006.

Pang B, Lee L. (2002) “Thumbs up? Sentiment Classification using Machine Learning Techniques”, *Proceedings of EMNLP*, 2002.

Pang B, Lee L. (2005) “Seeing Stars: Exploiting Class Relationships for Sentiment Categorization with Respect to Rating Scales”, *Proceedings of the 43rd Meeting of the ACL*, pp. 115-124, June 2005.

Pang B, Lee L. (2008) “Opinion Mining and Sentiment Analysis”, *Foundations and Trends in Information Retrieval*, Vol. 2, Nos. 1-2, pp. 1-135, 2008.

Pang B, Lee L. (2004) "A Sentimental Education: Sentiment Analysis Using Subjectivity Summarization Based on Minimum Cuts", *Proceedings of the ACL, 2004*.

Piatetsky-Shapiro G. (1999) “The Data Mining Industry coming of Age”, *IEEE Intelligent Systems*, pp. 32-34, 1999.

Piatetsky-Shapiro, G. (1991) “Knowledge Discovery in Real Databases: A Report on the IJCAI-89 Workshop. AI Magazine 11(5): 68–70, 1991.

Piatetsky-Shapiro, G. (2000) “Knowledge Discovery in Databases: 10 Years After”, *SIGKDD Explorations*, Vol. 1, No. 2, February 2000.

Piatetsky-Shapiro, G. (2007) “Data Mining and Knowledge Discovery 1996 to 2005: Overcoming the Hype and Moving from ‘University’ to ‘Business’ and ‘Analytics’”, *Data Mining and Knowledge Discovery*, Vol. 15, No. 1, 99-107, 2007.

Porter M.F, van Rijsbergen C.J., Robertson S.E. (1980) “New models in probabilistic information retrieval.*” London: British Library. (British Library Research and Development Report, no. 5587)*, 1980.

Provost F, Fawcett T. (2001) “Robust Classification for Imprecise Environments”, *Machine Learning*, Springer, Vol. 42, pp. 203-232, 2001.

Provost, J. (1999), "Naive-Bayes vs. Rule-Learning in Classification of Email", *Technical Report 99 - University of Texas*, 1999.

Prusak L, (2001) “Where did Knowledge Management Come From?”. *IBM Systems Journal*, Vol. 40, No 4, pp. 1002-1007, 2001.

Pyle D. (2004) “This Way Failure Lies”, *DB2 Magazine*, Issue 1, February 2004 [from http://www.ibmdatabasemag.com/story/showArticle.jhtml?articleID=17602328 ].

Quintas P, Lefrere P, Jones G. (1997) “Knowledge Management: A Strategic Agenda”, *Long Range Planning*, Vol. 30, No. 3, 385-391, 1997.

Ramaswamy, S. and Rastogi, R. and Shim, K. (2000) “Efficient algorithms for mining outliers from large data sets”, *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, pp. 427-438, 2000.

Read J. (2005) “Using Emoticons to reduce Dependency in Machine Learning Techniques for Sentiment Classification”, *Proceedings of the ACL Student Research Workshop*, 2005.

Rogati M, Tang Y. (2002) “High-performing feature selection for text classification”, *Proceedings of the ACM 11$^{th}$ international conference on Information and knowledge management*, pp. 659-661, 2002.

Rowley J. (2007) “The Wisdom Hierarchy: Representations of the DIKW hierarchy”, *Journal of Information Science*, Vol. 2, No. 33, 163-180, 2007.

Salton, G., Wong, A., Yang, C. S. (1975) "A vector space model for automatic indexing." *Communications of the ACM*, No. 18, Vol. 11 (Nov. 1975), 613-620, 1975.

Salton, G. and Buckley, C. (1987) "Term weighting approaches in automatic text retrieval", *Cornell University Technical Report*, 1987.

Salvetti F, Lewis S, Reichenbach C. (2004) “Automatic Opinion Polarity Classification of Movie Reviews”. *Colorado Research in Linguistics, June 2004, Volume 17, Issue 1. Boulder: University of Colorado.*, 2004.

Schneiderman B, (1996) “The Eyes Have it: A Task by Data Type Taxonomy for Information Visualization”, *IEEE Visual Languages*, 336-343, 1996.

Sebastiani F. (2002) “Machine Learnining in Automated Text Categorization”, *ACM Computing Surveys*, Vol. 34, pp. 1-47, 2002.

SEMMA, (2008) “SAS Enterprise Miner – Predictive Analytics / Data Mining” *SAS Institute Inc* , Accessed Nov/2008 from: [http://www.sas.com/technologies/analytics/datamining/miner/semma.html]

Shearer C. (2000) “The CRISP-DM Model: A New Blueprint for Data Mining”, *Journal of Data Warehousing*, Vol. 5, Number 4, Fall 2000.

Stavrianou, A., Andritsos, P., and Nicoloyannis, N. (2007). “Overview and semantic issues of text mining”. *SIGMOD Record*, Vol. 36, Sep. 2007, 23-34, 2007.

Strapparava C, Valitutti A, Stock O. (2006) “The affective weight of lexicon”, *Proceedings of the Fifth International Conference on Language Resources and Evaluation*, 2006.

Strapparava C, Valitutti A. (2004) “WordNet-Affect: an Affective Extension of Wordnet”, *Proceedings of the 4th International Conference on Language Resources and Evaluation*, 2004.

Sullivan S. (2005) “Search Engine Sizes”, *Search Engine Watch*, January 28th 2005, Accesed February 2009 from [http://searchenginewatch.com/showPage.html?page=2156481].

Swanson, D.R. and Smalheiser, N.R., (1997) “An interactive system for finding complementary literatures: a stimulus to scientific discovery”, *Artificial intelligence*, Vol. 91, No. 2, pp. 183-203, 1997.

Teece D, Pisano G, (2002) “The Dynamic Capabilities of Firms”, *Handbook on Knowledge Management,* 195-209, Springer, 2002.

Teece D. (1998) “Capturing Value from Knowledge Assets”, *California Management Review, Vol. 40, No. 3, Spring 1998.*

Tiwana A. (2002) “The Knowledge Management Toolkit – Practical Techniques for Building Knowledge Management Systems”, *Prentice Hall,* 2000.

Toutanova K, Manning C. (2000). “Enriching the Knowledge Sources Used in a Maximum Entropy Part-of-Speech Tagger”. *Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora (EMNLP/VLC-2000)*, pp. 63-70.

Turney P. (2002) “Thumbs up or Thumbs down? Sentiment Orientation Applied to Unsupervised Classification of Reviews”, *Proceedings of the 40th Annual Meeting of the Association of Computational Linguistics – ACL*, 2002.

van Rossum G, Drake F. (2003) “Python Language Reference Manual”, *Network Theory Ltd*., Bristol, UK, 2003.

von Krogh G, (1998) “Care in Knowledge Creation”, *California Management Review*, Vol 40, No 3, 133-153, Spring 1998.

Wang H, Wang S, (2008) “A Knowledge Management Approach to Data Mining for Business Intelligence”, *Industrial Management and Data Systems*, Vol. 108, No. 5, pp. 622-634, 2008.

Wei C, Piramuthu S, Shaw M. (2002) “Knowledge Discovery and Data Mining”, *Handbook on Knowledge Management*, 157-189, Springer, 2002.

Weiss S, Indurkhya N, Zhang T, Damerau F, (2005) “Text Mining – Predictive Methods for Analyzing Unstructured Information”. *Springer*, 2005.

Weiss S, Indurkhya N. (1998) “Predictive Data Mining – A Practical Guide”, *Morgan-Kauffman Publishers*, San Francisco, California, 1998.

Wiebe J, Bruce R, Martin M, Wilson T, Bell M. (2004) “Learning Subjective Language”, *Computational Linguistics*, Vol. 30, No. 3, pp. 277-308, January 2004.

Wiebe J, Bruce R, O’Hara T. (1999) “Development and Use of Gold-Standard Data Set for Subjectivity Classifications”, *Proceedings of the 37th Annual Meeting of the Association of Computational Linguistics – ACL-99*, pp. 246-253.

Wiebe J, Mihalcea R. (2006) “Word Sense and Subjectivity”, *Proceedings of the 21st International ACL Conference on Computational Linguistics*, pp 1065-1072, July, 2006.

Wiebe J. (1990) “Identifying Subjective Characters in Narrative”, *Proceedings of the 13th conference on Computational linguistics* – Vol. 2, Helsinki, Finland,

Wilks Y, Stevenson M. (1998) “Word Sense Disambiguation using Optimised Combinations of Knowledge Sources”, *Proceedings of the Annual Meeting of the Association of Computational Linguistics ACL*, Vol. 36, pp. 1398-1402. 1998.

Witten, I, Frank E. (1999) “Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations”, *Morgan Kaufmann*, San Francisco, 1999.

Wolpert D. (1996) “The Lack of a-priori Distinctions between Learning Algorithms”, *Neural Computation*, MIT Press, 1391-1421, 1996.

Wolpert D. (2001) “The Supervised Learning No-Free-Lunch Theorems”, *Proceedings of the 6th World Conference in Soft Computing*, 2001.

WORDNET (2006) “WordNet 3.0 Statistics”, *The WordNet Portal*, online at [http://wordnet.princeton.edu/man/wnstats.7WN] , accessed January 2009.

WORDNET (2009) “Wordnets in the world”, *WordNet Portal*, online at [http://www.globalwordnet.org/gwa/wordnet_table.htm], accessed January 2009.

XLMINER (2006) “XLMiner Capabilities”, *XLMiner Website*, Accessed Feb/09 from [http://www.resample.com/xlminer/capabilities.shtml], 2006.

Yang Y, Padmanabhan B. (2005), “GHIC: A hierarchical pattern-based clustering algorithm for grouping Web transactions” *IEEE Transactions on Knowledge and Data Engineering*, Vol.17, No. 9, pp. 1300-1304, 2005.

Yang Y, Pedersen J. (1997) “A Comparative Study in Feature Selection on Text Classification”, *International Workshop in Machine Learning – 1997*, pp. 412-420, 1997.

Yang, Y., Pedersen, J.O. (1997) "A Comparative Study on Feature Selection in Text Categorization", *Proc. of the 14th International Conference on Machine Learning ICML97*, pp. 412---420, 1997

Yu H, Hatzivassiloglou V. (2003) “Towards Answering Opinion Questions: Separating Facts from Opinions and Identifying Polarity in Sentences”, *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing*, pp. 129-136, 2003.

Zeleny M, (1987) “Management Support Systems: Towards Integrated Knowledge Management”, *Human Systems Management*, No. 7, 59-70, 1987

Zhang, G.P. (2000) “Neural Networks for Classification: a Survey”, *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, Volume 30, Issue 4, 451-462, November 2000.

# APPENDIX A – LANGUAGE RESOURCES

## *A.1 Penn TreeBank TagSet*

The Penn Treebank Tagset (Marcus et al, 1993).

Source: http://www.computing.dcu.ie/~acahill/tagset.html

| | |
|---|---|
| CC | Coordinating conjunction<br>**e.g.** and,but,or... |
| CD | Cardinal Number |
| DT | Determiner |
| EX | Existential *there* |
| FW | Foreign Word |
| IN | Preposision or subordinating conjunction |
| JJ | Adjective |
| JJR | Adjective, comparative |
| JJS | Adjective, superlative |
| LS | List Item Marker |
| MD | Modal<br>**e.g.** can, could, might, may... |
| NN | Noun, singular or mass |
| NNP | Proper Noun, singular |
| NNPS | Proper Noun, plural |
| NNS | Noun, plural |
| PDT | Predeterminer<br>**e.g.** all, both ... when they precede an article |
| POS | Possessive Ending<br>**e.g.** Nouns ending in 's |
| PRP | Personal Pronoun<br>**e.g.** I, me, you, he... |
| PRP$ | Possessive Pronoun |

| | |
|---|---|
| | **e.g.** my, your, mine, yours... |
| RB | Adverb<br>Most words that end in -ly as well as degree words like quite, too and very |
| RBR | Adverb, comparative<br>Adverbs with the comparative ending -er, with a strictly comparative meaning. |
| RBS | Adverb, superlative |
| RP | Particle |
| SYM | Symbol<br>Should be used for mathematical, scientific or technical symbols |
| TO | *to* |
| UH | Interjection<br>**e.g.** uh, well, yes, my... |
| VB | Verb, base form<br>subsumes imperatives, infinitives and subjunctives |
| VBD | Verb, past tense<br>includes the conditional form of the verb to be |
| VBG | Verb, gerund or persent participle |
| VBN | Verb, past participle |
| VBP | Verb, non-3rd person singular present |
| VBZ | Verb, 3rd person singular present |
| WDT | Wh-determiner<br>**e.g.** which, and *that* when it is used as a relative pronoun |
| WP | Wh-pronoun<br>**e.g.** what, who, whom... |
| WP$ | Possessive wh-pronoun |
| WRB | Wh-adverb<br>**e.g.** how, where why |

## *A.1 Stop Word List*

| | |
|---|---|
| a | of |
| about | on |
| an | or |
| are | that |
| as | the |
| at | this |
| be | to |
| by | was |
| for | what |
| from | when |
| how | where |
| in | who |
| is | will |
| it | with |
| la | the |

# APPENDIX B – PYTHON CODE

## *B.1 Negation Algortihm*

```
#
# populates array of negated terms based on document terms
# negation[i] indicates if term in doc[i] is negated
#
def getNegationArray(doc, windowsize):

 PSEUDO = ( 'no increase', 'no wonder', 'no change' , 'not cause' , 'not only' , 'not
necessarily' )
 PRENEGATION = ( 'not' , 'no' , 'n\'t' ,'cannot', 'declined' , 'denied' , 'denies' ,
'free of' , 'fails to' , 'no evidence' , 'no new' , 'no sign' , 'no suspicious' \
               'no suggestion' , 'rather than', 'with no' , 'unremarkable', 'without' ,
'rules out' , 'ruled out', 'rule out')
 POSNEGATION = ( 'unlikely', 'free', 'ruled out' )
 ENDOFWINDOW = ( '.', ':', ',', 'but' , 'however' , 'nevertheless' , 'yet' , 'though' ,
'although' , 'still' , 'aside from' , 'except' , 'apart from')

 # Initialise array
 vNEG = [ 0 for t in range(len(doc)) ]

 # Initialise window counters
 winstart = 0
 winend = min( windowsize, len(doc) - 1 )
 docsize = len(doc)

 i = 0
 found_pseudo = 0
 found_neg_fwd = 0
 found_neg_bck = 0
 inwindow = 0

 for i in range(docsize):

     #
     # build 1-ter and 2-term strings
     #
     unigram = doc[i].split('/')[0]
     if i < (docsize - 1):
         bigram = unigram + ' ' + doc[i+1].split('/')[0]
     else:
         bigram = unigram

     #
     # Search for pseudo negations
     #
     for negterm in PSEUDO:
      if bigram == negterm:
          found_pseudo=1
          ##print 'found pseudo!', bigram, i

     if (found_pseudo == 0):
         #
         # Look for pre negations
         #
         for negterm in PRENEGATION:
             if unigram == negterm or bigram == negterm:
                 found_neg_fwd = 1

         for negterm in POSNEGATION:
             if unigram == negterm or bigram == negterm:
                 found_neg_bck = 1
     #
     # If found fwd/backw negation, then negate window
     #
     if (found_neg_fwd == 1):
         ##print 'found forwards!', unigram, bigram, i
```

```
            #
            # negate terms forward up to window
            #
            if inwindow < windowsize:
                vNEG[i] = 1
                inwindow+=1
            else:
                # out of window space
                found_neg_fwd = 0
                inwindow = 0

        #
        # backward negation
        #
        if (found_neg_bck == 1):
            ##print 'found backwards!', unigram, bigram, i
            #
            # negate back until window start
            #
            for counter in range(max(winstart, i-windowsize), i):
                vNEG[counter] = 1

            #
            # done with backwards negation
            #
            found_neg_bck = 0

        #
        # now move window
        #
        for negterm in ENDOFWINDOW:
            if unigram == negterm or bigram == negterm:
                    #
                    # found end of negation, must reset windows
                    #
                    ##print 'found negterm!', unigram, bigram, i
                    inwindow = 0
                    found_neg_fwd = 0
                    winstart = i
                    winend = min( windowsize + i, len(doc) - 1 )

    return vNEG
```