

# Northumbria Research Link

Citation: Chalothorn, Tawunrat (2016) Quantitative Assessment of Factors in Sentiment Analysis. Doctoral thesis, Northumbria University.

This version was downloaded from Northumbria Research Link:  
<http://nrl.northumbria.ac.uk/30233/>

Northumbria University has developed Northumbria Research Link (NRL) to enable users to access the University's research output. Copyright © and moral rights for items on NRL are retained by the individual author(s) and/or other copyright owners. Single copies of full items can be reproduced, displayed or performed, and given to third parties in any format or medium for personal research or study, educational, or not-for-profit purposes without prior permission or charge, provided the authors, title and full bibliographic details are given, as well as a hyperlink and/or URL to the original metadata page. The content must not be changed in any way. Full items must not be sold commercially in any format or medium without formal permission of the copyright holder. The full policy is available online: <http://nrl.northumbria.ac.uk/policies.html>

[www.northumbria.ac.uk/nrl](http://www.northumbria.ac.uk/nrl)



# **QUANTITATIVE ASSESSMENT OF FACTORS IN SENTIMENT ANALYSIS**

Tawunrat Chalothorn

PhD

2016

# **QUANTITATIVE ASSESSMENT OF FACTORS IN SENTIMENT ANALYSIS**

Tawunrat Chalothorn

A thesis submitted in partial fulfilment  
of the requirements of the  
University of Northumbria at Newcastle  
for the degree of  
Doctor of Philosophy

Research undertaken in the  
Faculty of Engineering and Environment

June 2016

# Quantitative Assessment of Factors in Sentiment Analysis

Tawunrat Chalothorn

## Abstract

Sentiment can be defined as a tendency to experience certain emotions in relation to a particular object or person. Sentiment may be expressed in writing, in which case determining that sentiment algorithmically is known as sentiment analysis. Sentiment analysis is often applied to Internet texts such as product reviews, websites, blogs, or tweets, where automatically determining published feeling towards a product, or service is very useful to marketers or opinion analysts. The main goal of sentiment analysis is to identify the polarity of natural language text.

This thesis sets out to examine quantitatively the factors that have an effect on sentiment analysis. The factors that are commonly used in sentiment analysis are text features, sentiment lexica or resources, and the machine learning algorithms employed. The main aim of this thesis is to investigate systematically the interaction between sentiment analysis factors and machine learning algorithms in order to improve sentiment analysis performance as compared to the opinions of human assessors. A software system known as TJP was designed and developed to support this investigation.

The research reported here has three main parts. Firstly, the role of data pre-processing was investigated with TJP using a combination of features together with publically available datasets. This considers the relationship and relative importance of superficial text features such as emoticons, n-grams, negations, hashtags, repeated letters, special characters, slang, and stopwords. The resulting statistical analysis suggests that a combination of all of these features achieves better accuracy with the dataset, and had a considerable effect on system performance.

Secondly, the effect of human marked up training data was considered, since this is required by supervised machine learning algorithms. The results gained from TJP suggest that training data greatly augments sentiment analysis performance. However, the combination of training data and sentiment lexica seems to provide optimal performance. Nevertheless, one particular sentiment lexicon, AFINN, contributed better than others in the absence of training data, and therefore would be appropriate for unsupervised approaches to sentiment analysis

Finally, the performance of two sophisticated ensemble machine learning algorithms was investigated. Both the Arbiter Tree and Combiner Tree were chosen since neither of them has previously been used with sentiment analysis. The objective here was to demonstrate their applicability and effectiveness compared to that of the leading single machine learning algorithms, Naïve Bayes, and Support Vector Machines. The results showed that whilst either can be applied to sentiment analysis, the Arbiter Tree ensemble algorithm achieved better accuracy performance than either the Combiner Tree or any single machine learning algorithm.

## List of Contents

Abstract .....	III
List of Contents .....	IV
List of Figures .....	VIII
List of Abbreviations.....	IX
Acknowledgements .....	X
Declaration .....	XI
Chapter 1 : Introduction .....	1
1.1. Background .....	1
1.2. Motivation .....	2
1.3. Research Aims and Objectives.....	4
1.4. Research Contribution .....	5
1.5. Thesis Structure .....	5
1.6. Publications .....	7
Chapter 2 : Literature Review .....	9
2.1 Sentiment Analysis.....	10
2.1.1 Levels of sentiment analysis .....	10
2.1.2 Purposes of sentiment analysis .....	22
2.1.3 Processes to measure degree of sentiment .....	28
2.1.4 Comparative assessment of sentiment analysis.....	39
2.2 Machine Learning Algorithms .....	40
2.2.1 Natural language packages with machine learning capability .....	43
2.2.2 Real-world Application of machine learning.....	46
2.2.3 Sentiment analysis via single machine learning .....	49
2.3 Ensemble Learning Algorithms; Multiple Classifiers.....	52
2.3.1 Sentiment analysis via common methodology.....	52
2.3.2 Sentiment analysis via simple combining methodology.....	57
2.3.3 Sentiment analysis via meta-combining methodology .....	61
2.4 Conclusion.....	64
Chapter 3 : System Design.....	66
3.1 System User Interface .....	66

3.2	System Architecture .....	68
3.2.1	Data Input.....	68
3.2.2	Sentiment Lexicons.....	71
3.2.3	Combination.....	72
3.2.4	Pre-processing.....	72
3.2.5	Sentiment Resources .....	74
3.2.6	Supervised Learning Algorithms .....	75
3.2.7	Evaluation Method .....	78
3.2.8	Data Output .....	79
3.3	System Operation .....	80
3.4	Conclusion.....	81
Chapter 4 : Factorial Experiments in Sentiment Analysis .....		82
4.1	Factorial Experimental Design.....	82
4.2	Blocking Comparative Experimental Design.....	84
4.3	Experimental Design .....	86
4.4	Results and Analysis .....	89
4.4.1	Basic Analysis.....	90
4.4.2	ANOVA Analysis .....	92
4.5	Evaluation of SMS Data and Analysis .....	94
4.5.1	Basic Analysis.....	95
4.5.2	ANOVA Analysis .....	97
4.5.3	Correlation analysis of Twitter and SMS datasets .....	99
4.6	Conclusion.....	101
Chapter 5 : Novel Ensemble Experiments for Sentiment Analysis .....		103
5.1	Ensemble Learning Algorithms; Multiple Classifiers.....	103
5.2	Arbiter Tree .....	104
5.2.1	Implementation .....	107
5.3	Combiner Tree.....	113
5.3.1	Implementation .....	114
5.4	Analysis of Results.....	117
5.5	Discussion .....	122
5.6	Conclusion.....	125

Chapter 6 : Conclusions and Future Work .....	128
6.1 Thesis Summary .....	128
6.2 Research Questions and Answers.....	130
6.3 Thesis Limitations .....	133
6.4 Summary of Contributions to knowledge of this Thesis .....	133
6.5 Future Work .....	134
References .....	139
Appendices.....	157
Appendix I: Examples of sentiment that were expressed in the writing of poems, sonnets, histories, books and media .....	158
Appendix II: List of Stanford Part-Of-Speech Tagging .....	162
Appendix III: Simple Comparative Experimental Design .....	163
Appendix IV: Balance Incomplete Block Design .....	165
Appendix V: Ethical Issue and Approval Confirmation .....	167
Appendix VI: Twitter Ethical Issues .....	168
Appendix VII: Sample of Data Entry of RCBD ANOVA in SPSS .....	170
Appendix VIII: Comparison of means of the treatments .....	171
Appendix IX: Publications .....	173

## List of Tables

Table 2-1: Pattern tag table .....	11
Table 4-1: Data records of RCBD.....	85
Table 4-2: ANOVA for RCBD .....	85
Table 4-3: Example of RCBD data recorded in experiment .....	87
Table 4-4: The results of each feature analysed by using Naïve Bayes (F-score) .....	88
Table 4-5: The results from the Twitter dataset .....	89
Table 4-6: Tests of Factors and Treatments Effects of ANOVA of Twitter dataset .....	92
Table 4-7: Multiple comparison analysis of ANOVA of Twitter dataset.....	92
Table 4-8: Correlation analysis of ANOVA of Twitter dataset .....	93
Table 4-9: The results from SMS dataset.....	94
Table 4-10: Tests of Factors and Treatments Effects Two-ways ANOVA of SMS dataset .....	97
Table 4-11: Multiple Comparison analysis of ANOVA of SMS dataset.....	97
Table 4-12: Correlation analysis of ANOVA of SMS dataset.....	98
Table 4-13: Correlation analysis of Twitter and SMS datasets.....	100
Table 5-1: Comparison of mean of the top five treatments .....	108
Table 5-2: The results of Arbiter and Combiner Trees .....	117
Table 5-3: Ranking data in the Wilcoxon signed-ranks test .....	120
Table 5-4: Output of Arbiter Tree from Wilcoxon signed-ranks test .....	121
Table 5-5: The results of Stacking .....	122
Table 5-6: Output of the comparison of Arbiter Tree and SVM from Wilcoxon signed-ranks test .....	123
Table 5-7: The results of Stacking .....	124
Table 5-8: Example of the two voters with the conditions .....	124
Table 5-9: The results of Majority Voting .....	125
Table 6-1: Example of dataset that has both discrete and continuous polarity ...	136



## List of Figures

Figure 2-1: Example of webpage with advertising's contents .....	24
Figure 2-2: Hyperplane of support vector machine .....	42
Figure 2-3: Example of ARFF file .....	46
Figure 3-1: User interface of TJP system .....	67
Figure 3-2: Simple UML class diagram of TJP system .....	67
Figure 3-3: Example of Tweet data formats that were received .....	70
Figure 3-4: Example of SMS data format that were received .....	71
Figure 3-5: Example of sentences that used Denecke (2008) methods .....	75
Figure 3-6: Example of data format in SVMLight .....	76
Figure 3-7: Example of data output from the Tweets .....	79
Figure 3-8: The operation of TJP system .....	80
Figure 4-1: Decision tree diagram of factorial design .....	83
Figure 4-2: Declare factors and response variable .....	87
Figure 4-3: Visualisation of F-score from Twitter dataset .....	90
Figure 4-4: Visualisation of F-score from SMS dataset .....	95
Figure 4-5: Q-Q plots of data from Twitter dataset .....	99
Figure 4-6: Q-Q plots of data from SMS dataset .....	100
Figure 5-1: Theory flowchart to produce a training dataset for Arbiter Tree .....	104
Figure 5-2: Sample of training data generated by selection rules .....	106
Figure 5-3: Theory flowchart of how the final prediction is made in Arbiter Tree .....	107
Figure 5-4: The overall of Arbiter Tree pipeline in the TJP system .....	108
Figure 5-5: Sample of training data generated by selection rules for using in TJP system .....	110
Figure 5-6: Diagram of Arbiter Tree in TJP system .....	111
Figure 5-7: Sample of training data that generated by composition rules .....	114
Figure 5-8: Theory flowchart of how the final prediction is made in Combiner Tree .....	114
Figure 5-9: The overall of Combiner Tree pipeline in the TJP system .....	115
Figure 5-10: Diagram of Combiner Tree in TJP system .....	116
Figure 5-11: Histogram plots of data from Arbiter Tree from R .....	118
Figure 5-12: Q-Q plots of data from Arbiter Tree .....	118
Figure 5-13: Flowchart for choosing the appropriate statistical test .....	119

## List of Abbreviations

Abbreviation	Description and Resource
AFINN (Finn Årup Nielsen's Lexicon)	List of English words created by using the contents from microblogs. <i>Resource:</i> Nielsen (2011a), ss. 2.1.3.7
ANOVA (Analysis of Variance)	Statistic method use to analyse the differences between more than two population mean. <i>Resource:</i> Chatfield (1983a); Montgomery and Runger (2007); Montgomery (2013a), ss. 4.2.1
BIBD (Balanced Incomplete Block Design)	An incomplete block design, whereby not all treatments are present in every block. <i>Resource:</i> Montgomery (2013a), ss. 4.2
HL (Hu Liu Lexicon)	Collection of online customer product reviews. <i>Resource:</i> Hu and Liu (2004), ss. 2.1.3.7
MaxEnt (Maximum Entropy Modelling)	Flexible feature-based model that aims to satisfy the constraints of available information. <i>Resource:</i> Harte (2011), ss. 2.2
MPQA (MPQA Subjective Lexicon)	Collection of news articles. <i>Resource:</i> Wilson <i>et al.</i> (2005b), ss. 2.1.3.7
RCBD (Randomized Complete Block Design)	Extension of the paired t-test (dependent t-test) to a situation where the factor of interest has more than two levels; that is, more than two treatments must be compared. <i>Resource:</i> Montgomery (2013a), ss. 4.2
SPSS (IBM SPSS statistic software )	Statistic software from IBM. <i>Resource:</i> IBM (2010), ss. 4.4.2
SVM (Support Vector Machine)	Binary linear classification model with the learning algorithm for classification and regression analysing the data and recognising the pattern. <i>Resource:</i> Kecman (2005), ss. 2.2
SWN (SentiWordNet)	Result of automatic annotation of all the WordNet synsets. <i>Resource:</i> Baccianella <i>et al.</i> (2010a), ss. 2.1.3.6
SS (SentiStrength)	Sentiment analysis methodology used to judge whether a sentence has a positive or negative sentiment. <i>Resource:</i> Thelwall <i>et al.</i> (2010b), ss. 2.1.3.6
TR (Training data)	Training dataset from SemEval 2013 Task 2A. <i>Resource:</i> Wilson <i>et al.</i> (2013), ss. 4.4

## **Acknowledgements**

I would like to thank you my parents who sponsored my research and stay in the UK. I owe lots to my parents for everything they done for me.

I also thank Jirapon Tanasanti, my brother, for all the support and patience he has shown me.

I also thank Dr. Jeremy Ellman and Dr. Paul Vickers, my supervisors, for their suggestions, guidance and near infinite patience.

## **Declaration**

I declare that the work contained in this thesis has not been submitted for any other award and that it is all my own work. I also confirm that this work fully acknowledges opinions, ideas and contributions from the work of others.

Any ethical clearance for the research presented in this thesis has been approved. Approval has been sought and granted by the Faculty Ethics Committee / University Ethics Committee / external committee on [**date**].

I declare that the Word Count of this Thesis is 0000 words

Name:

Signature:

Date:

## Chapter 1 : Introduction

### 1.1. Background

Emotions are compound feelings concerned with a person and or an object which tends to be both intense and focused (Izard, 1991; Liu, 2015). When emotions are expressed in written form the linguistic term ‘sentiment’ is preferred to distinguish the mental state from its expression. Sentiment can be defined as a tendency to experience certain emotions in relation to a particular object or person. Usually, sentiments are expressed in many written forms, such as poems, sonnets, histories, books and media. (See Appendix I for an example of each one). Sentiments are frequently hidden within long sentences or displayed as idioms; thus rendering them more difficult to read and extract.

There is a field of research in natural language processing (Hogenboom *et al.*, 2012) called, sentiment analysis. Sentiment analysis may also be referred to as opinion mining, which is the study of people’s opinions, appraisals and emotions towards entities, events and their attributes (Pang and Lee, 2008).

Over the last decade sentiment analysis has received attention within several research areas; such as marketing and production (Mishne and Glance, 2006; Grabner *et al.*, 2012), political organisations (Tumasjan *et al.*, 2010), psychology (Hobson *et al.*, 1998; Domhoff, 2003; St-Onge *et al.*, 2005). This period has also been distinguished by the rapid development of internet technologies, leading to their easy availability and mass exploitation. These factors enabled the considerable growth in internet users who create vast amounts of data each day.

User-generated content is a valuable source of information as it contains people’s opinions and judgements on a topic. The basic task of sentiment analysis is to classify the polarity of a given text. This is known as sentiment classification (Pang *et al.*, 2002; Turney, 2002; Thelwall *et al.*, 2010a; Troussas *et al.*, 2013; Balahur, 2013). The main goal of sentiment classification is to identify the polarity of natural language text. The majority of research on sentiment classification considers this to be a binary problem, where a text has either a positive or negative polarity (Ponomareva, 2014).

The motivation for working in the area of sentiment analysis is presented in the following section.

## **1.2. Motivation**

Social networking has become pervasive in our society. The simplicity of the Internet has enabled users to post their thoughts and sentiments in a variety of diverse forms, many of which remain largely unmonitored. For instance, blogging is particularly rich in sentiment and read daily by millions of web users. This has led to blogs being regarded as the latest form of self-expression and it is possible to track specific discussion threads over several months.

There is research that classifies customer reviews through the use of blogs and websites. For example, Pang *et al.* (2002) classified movie reviews by using supervised learning algorithms. Hu and Liu (2004) analysed product reviews by using feature-based sentiment analysis. Popescu and Etzioni (2007) used unsupervised learning algorithm to identify features and opinions from customers' reviews. Hu *et al.* (2012) used sentiment analysis to detect users' opinions of books, whilst Duan *et al.* (2013) analysed hotel service quality by using the Naïve Bayes machine learning algorithm (Tan *et al.*, 2009).

Currently, the micro-blogging tool Twitter is well-known and increasingly popular. Twitter allows its users to post messages, or 'Tweets' of up to 140 characters each time, which are available for immediate download over the Internet. Tweets are interesting to marketers since their rapid public interaction can either indicate customer success or presage public relations disasters far more quickly than web pages or traditional media. There is research that has used Twitter to analyse customers' reviews. For example, Jmal and Faiz (2013) used Twitter trends to measure customer satisfaction towards products such as digital cameras, phone and iPod, and used in the classification. Gautam and Yadav (2014) classified the Tweets dataset and claimed that they made the contribution to used sentiment analysis classification of customers' reviews.

From these articles, the questions arose, "how could sentiment analysis be further used to analyse customers' reviews from Twitter?" To answer this question, we have to start from a quantitative assessment of the factors required

for sentiment analysis. Consequently, we participated in an international competition on Sentiment Analysis, SemEval 2013 task 2A (Wilson et al., 2013). This allowed us to consider the importance of factors and sentiment analysis within the scope of a dataset which was used by multiple research groups. SemEval 2013 task 2A (Wilson et al., 2013) itself was intended to promote the research area of sentiment analysis, with a view to obtaining a better understanding of how sentiment is taken in contexts using the Twitter sentiment corpus (Wilson et al., 2013). The dataset is made up of Tweets and SMS text (Wilson et al., 2013). The Tweets were collected from Twitter over one-year period spanning from January 2012 to January 2013 by using the Twitter API (Wilson *et al.*, 2013). For SMS data, SemEval 2013 Task 2A (Wilson *et al.*, 2013) used the data from the NUS SMS corpus (Chen and Kan, 2013). Tweets were used as training data (8852 lines), testing data (3558 lines) and gold standard<sup>1</sup> (3558 lines). The SMS were used as testing data (2175 lines) and as a gold standard (2175 lines). The purpose for having SMS is to observe how generalizable a system trained on Tweets may be for the other types of data.

Both Tweet and SMS datasets contain marked instances of words or phrases whose sentiment was to be determined. The boundaries for the marked instance were provided. Both Tweet and SMS datasets were annotated using the Amazon Mechanical Turk<sup>2</sup>. Each sentence was marked up by five human annotators using the start and end point of their opinion for the phrase or word. They then stated whether this phrase or word had negative, neutral or positive sentiment.

There are 2 subtasks in task 2A: constrained and unconstrained. The constrained task uses the training data provided only; other resources, such as lexicons were allowed. The unconstrained task uses the training data a provided and additional data for training, such as additional tweets/SMS messages or additional sentences annotated for sentiment. This thesis considers the constrained task by using the original training data without using any other resources. This allowed the exploration of the relative success of a simple approach of machine

---

<sup>1</sup> The gold standard is especially important as it refers to the testing data whose polarity is labelled by human annotators, and is assumed to be correct. This will be used to measure the accuracy of the experiments reported here.

<sup>2</sup> Amazon Mechanical Turk is an internet marketplace service for work that requires human intelligence.

learning by using dataset that was given from organiser without any additional data.

44 teams took part in SemEval 2013 task 2A (Wilson et al., 2013), who used a total of 149 different techniques and achieved different accuracy scores. It would therefore be useful to identify the factors that impacted on the task, whether they be sentiment resource or software, and how the accuracy may be improved by using a combination of the factors within an ensemble learning algorithm. A software system was designed so that the factors within sentiment analysis could be selected and modified in comparison and evaluated to determine the possible outcome. Consequently, the final scope of this thesis was determined after participating in SemEval 2013 Task 2A (Wilson et al., 2013).

### **1.3. Research Aims and Objectives**

In the previous section, the motivation is given for the construction of sentiment analysis. Further investigation of sentiment analysis showed that there are different approaches, although it is not clear how to determine which factors were appropriate in the collaboration. People have previously tried different approaches and there is no systematic comparison between the effectiveness of the different factors. Consequently, the aim is to investigate and identify the factors that are important and have the most significant effect on sentiment analysis. In order to achieve these aims, the following main objectives are established:

1. Research several classifiers such as Naïve Bayes (Tan *et al.*, 2009), Support Vector Machine (SVM) (Kecman, 2005) and Maximum Entropy Modelling (MaxEnt) (Harte, 2011) and factors that are used commonly in sentiment analysis. The factors may include: feature(s), dataset, sentiment lexicon(s), and sentiment resource.
2. Perform a comparison of several classifications and factors applied within the same environment.
3. Evaluate and rank the results from object 2 by aggregating each classification in terms of the factors.



4. Investigate the existing methods in ensemble learning algorithms that have not been used in sentiment analysis.
5. Apply the top results from objective 3 and evaluate the results.

#### **1.4. Research Contribution**

This thesis makes three contributions to sentiment analysis. These are the determination and classification of the expression of features and identifying the other relevant factors. The purpose of these contributions are to investigate the features within pre-processing data through showing them as feature matrixes and investigating them through factorial experiments concerning the feature's effectiveness in sentiment analysis performance. In other words, we attempted to identify which factor(s) brought the most significant improvement to system performance. To determine and classify the expression of features, eight features were used: emoticons, n-gram, negations, Twitter features, repeated letters, special characters, slang and stopwords.

Finally, we propose and perform a process to re-contextualise the existing methods within ensemble learning that have not been used previously in sentiment analysis. These are the Arbiter Tree (Chan and Stolfo, 1993) and Combiner Tree (Chan and Stolfo, 1997) ensemble machine learning algorithms. We investigated and demonstrated how they can be applied to sentiment analysis. We conclude that it is worth trying the Arbiter Tree (Chan and Stolfo, 1993) method in sentiment analysis.

#### **1.5. Thesis Structure**

The remainder of this thesis is structured as follows:

**Chapter 2: Literature Review.** This chapter discusses previous studies conducted in the field of sentiment analysis, machine learning and ensemble learning. Relevant works of sentiment analysis are categorised according to how they are used to measure the degree of sentiment; for example, using polarity of words, human annotators, emoticons, feature-based, range of polarity and sentiment resources and sentiment lexicons. The works that relate to machine

learning algorithms are presented as a category of single machine learning. Finally, the works that relate to ensemble learning are categorised using two types of methodology; common and combining methodologies. Due to the focus of the thesis, special attention is paid to the studies investigating the relation between the sentiment lexicon and single machine learning algorithms and combining the methodology in ensemble learning algorithms.

**Chapter 3: System Design.** This chapter gives the design of the TJP system which is used to carry out factorial experiments in sentiment analysis. The system was designed so that each possible factor in the analysis could be turned on or off, allowing the experiment to be carried out with or without the individual factors. The factors are composed of dataset, sentiment lexicons, sentiment resources, single machine learning algorithms and ensemble learning algorithms. The system's results are then used to carry out the factorial experiment.

**Chapter 4: Factorial Experiments in Sentiment Analysis.** Experiments that study the effects of one or more factors are known as factorial experiments. Factorial experimental design is an area of statistics that impacts on experimental disciplines such as psychology or agriculture, where possible combinations of factor levels are investigated (Montgomery, 2013b). Therefore, this chapter described a series of systematic experiments whose aim was to identify the relative importance of different factors in sentiment analysis. In the factorial experiment, a repeated measures design was used because there are three machine learning algorithms (independent variables). The machine learning algorithms are within-subject and tested as subject variable. Each subject was tested using each level of the variables, which are training datasets, lexicons and a combination of training datasets and lexicons. Moreover, they are analysed using randomised complete block designs due to all the blocks in the experiment being filled without missing any treatments.

**Chapter 5: Novel Ensemble Experiment for Sentiment Analysis.** This chapter is concerned specifically with sentiment analysis using techniques in ensemble

learning algorithms. Ensemble learning is an approach of machine learning algorithms that uses multiple classifiers to train data and make the final prediction, which often achieves a higher accuracy than using a single classifier. This is considered as novel, as after reviewing the related work in the area of sentiment analysis, we found that neither the Arbiter Tree (Chan and Stolfo, 1993) nor Combiner Tree (Chan and Stolfo, 1997) methods had been used in sentiment analysis. Therefore, they are investigated, implemented and applied within the new context (micro-blogging and short message) to test the theories in a new setting (sentiment analysis) and showed whether they work or not. This chapter concludes with a discussion of the comparison of using those theories and others that are used commonly in sentiment analysis. The results are analysed using a non-parametric method.

**Chapter 6: Conclusion.** This chapter critically assesses the techniques and experiments of the work reported in this thesis. The contributions of this thesis are outlined and discussed. Finally, some recommendations for future work are proposed.

### 1.6. Publications

The publications concerned with this thesis are presented below:

**Conference Paper:** CHALOTHORN, T. & ELLMAN, J. 2012. Using SentiWordNet and Sentiment Analysis for Detecting Radical Content on Web Forums. *The 6th Conference on Software, Knowledge, Information Management and Applications (SKIMA 2012)*. Chengdu, China.

**Conference Paper:** CHALOTHORN, T. & ELLMAN, J. 2012. Sentiment Analysis Of Web Forums: Comparison Between SentiWordNet And SentiStrength. *The 4th International Conference on Computer Technology and Development (ICCTD 2012)*. Bangkok, Thailand.

**Conference Paper:** CHALOTHORN, T. & ELLMAN, J. 2013. TJP: Using Twitter to Analyze the Polarity of Contexts. *Second Joint Conference on Lexical and Computational Semantics (\*SEM), Volume 2: Proceedings of the Seventh*

*International Workshop on Semantic Evaluation (SemEval 2013)*. Atlanta, Georgia, USA: Association for Computational Linguistics.

**Conference Paper:** CHALOTHORN, T. & ELLMAN, J. 2013. Sentiment Analysis: State of the Art *Proc. of the Intl. Conf. on Advances in Computer and Electronics Technology (ACET 2013)*. Hong Kong: UACEE.

**Conference Paper:** CHALOTHORN, T. & ELLMAN, J. 2014. TJP: Identifying the Polarity of Tweets from Contexts. *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*. Dublin, Ireland: Association for Computational Linguistics and Dublin City University.

**Conference Paper:** CHALOTHORN, T. & ELLMAN, J. Using Arbiter and Combiner Tree to Classify Contexts of Data. International Conference on Computer and Information Technology (ICCIT 2015), 2015 Ankara, Turkey.

**Conference Paper:** CHALOTHORN, T. & ELLMAN, J. Simple approaches of sentiment analysis via ensemble learning. The International Conference on Information Science & Applications (ICISA 2015), 2015 Pattaya, Thailand.

**Journal Paper:** CHALOTHORN, T. & ELLMAN, J. 2013. Affect Analysis of Radical Contents on Web Forums Using SentiWordNet. *International Journal of Innovation, Management and Technology (IJIMT)*, 4, 122.

**Journal Paper:** CHALOTHORN, T. & ELLMAN, J. 2015. Two classifiers in Arbiter Tree to analyse data. *International Journal of Advances in Engineering and Technology (IJAET)*, Vol. 7, pp. 1657.

## Chapter 2 : Literature Review

This chapter focuses on the three aspects that are related to this research, namely, sentiment analysis, machine learning algorithms and ensemble learning algorithms. In sentiment analysis (Section 2.1), the details of level, purposes and processes that used for measuring degree of sentiment are briefly described in Sections 2.1.1, 2.1.2 and 2.1.3, respectively. Moreover, the details of SemEval 2013 (Wilson *et al.*, 2013) which we participated are mentioned in Section 2.1.4

For machine learning algorithms, there are three single algorithms that we interested. They are Naïve Bayes (Tan *et al.*, 2009), Support Vector Machine (SVM) (Kecman, 2005) and Maximum Entropy Modelling (MaxEnt) (Harte, 2011).. They are chosen because they were used the most in SemEval 2013 (Wilson *et al.*, 2013). They details are briefly discussed in Section 2.2, followed by details of popular natural language packages that contain the abilities of machine learning algorithms in Section 2.2.1. The real-world applications that used machine learning algorithms are sampled in Section 2.2.2. Some related work of sentiment analysis that used machine leaning algorithms are discussed in Section 2.2.3.

For ensemble learning algorithms can be separated to three families. They are common methodology (Section 2.3.1), simple combining methodology (Section 2.3.2) and meta-combining methodology (Section 2.3.3). There are two major algorithms in meta-combining methodology considered in this thesis. They are Arbiter Tree (Chan and Stolfo, 1993) and Combiner Tree (Chan and Stolfo, 1997). These algorithms were chosen since it has been claimed that they can be used in sentiment analysis (Rokach, 2005; Rokach, 2009; Rokach, 2010). However, no studies or related work have used either the Arbiter Tree (Chan and Stolfo, 1993) or Combiner Tree (Chan and Stolfo, 1997) in sentiment analysis. Moreover, we would like to know if they will face the same data sensitivity problem that the other algorithms faced. For example, Martin-Valdivia *et al.* (2013) (Section 2.3.3) presented evidence that the results from Stacking (Wolpert, 1992) achieved slightly higher results than Majority Voting (Polikar, 2012). On the other hand, Gryc and Moilanen (2014) (Section 2.3.2) found that Majority

Voting (Polikar, 2012) achieved higher results than Stacking (Wolpert, 1992). Their results showed that the performance of Stacking (Wolpert, 1992) and Majority Voting (Polikar, 2012) are changed when the datasets were changed which is the issue of data sensitivity.

## **2.1 Sentiment Analysis**

Sentiment can be defined as a tendency to experience certain emotions in relation to a particular object or person. Sometimes, opinions are hidden within long sentences, making them difficult to read and extract. The approach known as ‘sentiment analysis’ is an aspect of Natural Language Processing (NLP). NLP is a research area that explores and analyses how the natural language text entered via a computer can be manipulated and transformed into a form more suitable for further processing (Chowdhury, 2010). Sentiment Analysis is the process of identifying sentiment from the written text. Such texts may be in the form of a document, paragraph, sentence or word length. Moreover, sentiment analysis is common in text-based electronic media, such as product reviews, websites, blogs, forums, etc. The main goal of sentiment analysis is to identify the polarity of text. That is, the expressed or implied emotional relationship of the text’s author to its subject. Polarities are not limited to being positive and negative (Section 2.1.1.4).

Sentiment Analysis may be referred to as ‘Opinion Mining’ as both study people’s opinions, appraisals and emotions towards entities, events and their attributes (Pang and Lee, 2008). As such, the area is of considerable interest to marketing, whose practitioners wish to identify public attitudes towards products, companies, political parties etc. We will now proceed to describe works around sentiment analysis applied to texts of varying lengths and for differing purposes.

### **2.1.1 Levels of sentiment analysis**

The analysis of sentiment classification can be performed at four levels: word-level, phrase-level, sentence-level and document-level. Brief details of each level are given in the following section.

### 2.1.1.1 Document-level sentiment analysis

Document-level analysis determines the sentiment of a whole document; for example, news, reviews, forums, blogs, or longer texts. Product reviews are an especially interesting domain area, as the text in the review can be evaluated against the review author's own opinion as expressed with a thumbs up or down. For example, Turney (2002) collected 410 reviews from the general consumer review site Epinions.com. These covered four different topic areas: automobiles, banks, movies and travel destinations. Turney (2002) classified these using an unsupervised algorithm, a learning algorithm that does not require labelled data as the input, at both document level and phrase level, and then evaluated his classification against the reviewer's thumbs up or down choices.

There are three steps in Turney (2002) system. Firstly, the reviews were analysed to identify whether the phrase contained adjectives or adverbs. This was done using the Brill (1994), a part-of-speech tagger. Part-of-speech can be defined as the grammar article class that words should be placed into, according to the work they do within a sentence, such as nouns, verbs, adjectives and adverbs. Two consecutive words were extracted from the reviews if their tags conformed to any of a predetermined set of patterns, as shown in the table reproduced in Table 2.1 (Turney, 2002).

First Word	Second Word	Third Word (Not Extracted)
1 JJ <sup>3</sup>	NN <sup>4</sup> or NNS <sup>5</sup>	anything
2 RB <sup>6</sup> , RBR <sup>7</sup> , or RBS <sup>8</sup>	JJ	Not NN nor NNS
3 JJ	JJ	Not NN nor NNS
4 NN or NNS	JJ	Not NN nor NNS
5 RB, RBR or RBS	VB <sup>9</sup> , VBD <sup>10</sup> , VBN <sup>11</sup> , or VBG <sup>12</sup>	anything

**Table 2-1:** Pattern tag table  
(Turney, 2002).

<sup>3</sup> Adjective

<sup>4</sup> Noun, singular or mass

<sup>5</sup> Noun, plural

<sup>6</sup> Adverb

<sup>7</sup> Adverb, comparative

<sup>8</sup> Adverb, superlative

<sup>9</sup> Verb, base form

<sup>10</sup> Verb, past tense

<sup>11</sup> Verb, past participle

<sup>12</sup> Verb, gerund or present participle

Secondly, an algorithm for estimating the semantic orientation of the phrases was used, where semantic orientation is a prediction method that refers to the positive or negative semantic and the degree to which the semantic of the text is carried (Butler, 2010). Semantic orientation can be calculated using the degree to which the word is associated with positive words minus its association with negative words (Butler, 2010). The ‘Pointwise Mutual Information and Information Retrieval’ algorithm (PMI-IR) (Turney, 2001) was used to evaluate the semantic orientation of extracted phrases (Hatzivassiloglou and McKeown, 1997b). For example, if a phrase has positive associations such as ‘romantic ambience’, the semantic orientation of the phrase will be positive. Conversely, if the phrase has negative associations, such as with horrific events (e.g. ‘train wreck’), the semantic orientation of the phrase will be negative.

Turney (2002) final step was to determine whether or not the reviews are recommendations. This is done by computing the average semantic orientation of phrases extracted from the reviews. If the average scores are positive, the review classification is ‘recommended’, and vice versa for the negative scores. The overall results achieved at 74.39% accuracy of the classification with the star rating.

In 2013, Moraes *et al.* (2013) used document-level sentiment classification in the empirical comparison between SVM (Kecman, 2005) and Artificial Neural Networks (ANN, (White, 1989). Artificial Neural Networks (White, 1989) are a biologically inspired computation model based on the structure and function of brain neural structures (Gershenson, 2003). Four datasets were used. They are the benchmark Movie review dataset from (Pang and Lee, 2004) and the collection of reviews from Amazon.com that was focused on GPS, Books and Cameras. The collection of these reviews were assigned the labelled by using the stars. The reviews were defined as positive, if the reviews contained more than 3 stars. The reviews that contained fewer than 3 stars were defined as negative. The reviews that contained exactly 3 stars were negative have not been included in the datasets. The data was passed to pre-processing to remove stopwords and stemmed before being used with SVM (Kecman, 2005) in LibSVM (Chang and Lin, 2011) (Section 2.1.1.4) and ANN (White, 1989) in Matlab (Matlab, 1994).



Before use the dataset was separated into two groups for use with balanced and unbalanced classifiers. The balance data referred to the number of data that was labelled as positive and negative classes are equal. In contrast, the unbalanced referred to the number of data in both classes are not equal. Moraes *et al.* (2013) reported the results of unbalanced and balanced data that ANN outperformed SVM in 13 tests out of 28 tests, although, the accuracy difference between ANN and SVM did not exceeded 3%.

Besides these two examples of document-level sentiment analysis, document-level sentiment classification has also been used in the other approaches of sentiment analysis based on sentence-level and word-level. The details of sentence-level are briefly described in the following section while the details of word-level are in Section 2.1.1.4.

#### **2.1.1.2 Sentence-level sentiment analysis**

A sentence-level consists of two main tasks. The first task is to classify whether the sentences are subjective or objective. ‘Subjective’ refers to the opinion expressions that describe people’s sentiments or feelings toward entities. In contrast, the entities, events and their properties are referred to as ‘objective’ (Liu, 2010). The second task is to classify the polarity of subjective sentences.

For example, Yu and Hatzivassiloglou (2003) used both document level and sentence level sentiment analysis to classify the opinion from the answers to questions. The articles were selected randomly from the collection of Newswire articles, focusing on editorial, business and news. The articles were separated and labelled into three groups: fact, opinion and uncertain. However, only fact and opinion labels were used. Additionally, there were three parts to the classification process.

The document level is the first part of the process in which the whole documents were trained with Naïve Bayes (Tan *et al.*, 2009) (Section 2.2). The sentence level is the second part, where the semantic oriental (Section 2.1.1.1) method was used to classify the contents’ polarities. Four features were used: words, bigrams, trigrams and part-of-speech for each sentence. Moreover, the presence of positive and negative words in sentences was an indicator of sentence

subjectivity (Hatzivassiloglou and Wiebe, 2000). The overall results achieved a 97% accuracy performance.

Meena and Prabhakar (2007) extracted sentiment from reviews. The datasets comprised of 20,000 pre-labelled, positive and negative sentences. Meena and Prabhakar (2007) wrote that their experiment did not use training data, but instead used lexicons. The sentences were passed to Lex-Parser (The Stanford Natural Language Processing Group, 2002) to collect the grammatical structure of sentences (see Appendix II for a table of the Stanford parser); whereby the output is the dependency tree, which is a directed acyclic graph with words as nodes and relations as edges (Ambati, 2008), with part-of-speech tagging and the types of dependencies of the word (number of the word order).

To determine the polarity of words, Meena and Prabhakar (2007) used General Inquirer (GI) (Stone *et al.*, 1968), which is a dictionary program used to determine a word's orientation. If a word does not exist in GI (Stone *et al.*, 1968), the database of English words, called WordNet (Fellbaum, 2010; Princeton University, 2010) was used instead. Next, the conjunction rules were applied to analyse the effectiveness of conjunctions which are words that are used to link words, phrases or clauses and may be used to indicate the relationship between the ideas expressed in the sentence (Meena and Prabhakar, 2007).

For example, everyone/NN but/CC John/NN is /VBZ present/JJ, the polarity for the right, NN will be opposite of the polarity of the left NN. Therefore, as the sentence is positive towards everyone, it is negative toward John and this is what the rule describes. Finally, once each word/phrase has its polarity, the overall sentiment of sentences is determined based on the comparison of tags and conjunction rules. The results from the sentences with conjunctions showed that Meena and Prabhakar (2007) achieved better accuracy from using conjunction analysis of GI (Stone *et al.*, 1968) with WordNet (Fellbaum, 2010; Princeton University, 2010) (Section 2.1.1.2) at 78% than using just GI (Stone *et al.*, 1968), which achieved an accuracy of 62%.

### 2.1.1.3 Phrase-level sentiment analysis

This level involves the classification of the polarity of phrases, such as a noun phrase<sup>13</sup> and verb phrase<sup>14</sup>. For example, the phrase level was used to extract the semantic orientations of newspaper articles by Takamura *et al.* (2007). The articles were extracted pairs of a noun and an adjective. They were annotated with semantic orientation (Section 2.1.1.1) tags and labelled as positive, neutral and negative. The Potts model (Wu, 1982) was adopted to extract the semantic orientation of phrasal expressions based on the idea from the Ising model (Gallavotti, 1972) that is concerned with transition that occurs when a small change occurs in a parameter (Cipra, 1987). The average classification accuracy of the phrase was obtained using a 10-fold cross-validation. A 10-fold cross-validation is the method where datasets were split into 10 sets with the size divided by 10. 9 datasets will be used as training data and 1 will be used as the validate data for testing the model. The methods were repeated 10 times before taking an indication of accuracy (Poole and Mackworth, 2010). The results showed that the accuracy obtained was 90.76%, 81.75% and 86.85% for positive, neutral and negative, respectively.

Tan *et al.* (2011) generated the typed dependencies of datasets using the Stanford parser (Section 2.1.1.2) before detecting the polarity of a phrase by using an algorithm that was adopted from Liu (2007), called the Class Sequential Rules (CSR) (Hu and Liu, 2006). CSR (Hu and Liu, 2006) is an algorithm that is used to generate the language patterns and is different to a classic sequential pattern because CSR (Hu and Liu, 2006) has a fixed target (class) (Hu and Liu, 2006).

The typed dependencies and polarity tagged bigram words were manually annotated using two human annotators. Three polarities were used: positive, neutral and negative. Tan *et al.* (2011) used three types of type dependency: adjectival modifier (AMOD), adverbial modifier (ADVMOD) and direct object modifier (DOBJ). The agreement of annotators was measured using a statistical method called Cohen's kappa (Cohen, 1968). Cohen's kappa (Cohen, 1968) was 0.78, which is considered acceptable. In contrast, each word in the typed

---

<sup>13</sup> Noun phrase is a phrase that has noun or pronoun as the head word; for example, it is pink. 'it' is a noun phrase of the sentence.

<sup>14</sup> Verb phrase is a part of the sentence that contains both verb and object; for example, TC has finished her lunch. 'finished her lunch' is a verb phrase of the sentence.

dependency polarity tagged bigram lexicon were assigned polarity by using lexicons from Thet et al. (2010) and Wilson *et al.* (2009).

The performance of CSR (Hu and Liu, 2006) is compared with the modified heuristic rules (Thet *et al.*, 2010), which is the conceptual data modelling that is often guided by common sense. However, the rules of heuristic rules depend on the researchers adjusting and developing the rules (Du, 2008). See Tan *et al.* (2011) for the full table of heuristic rules that were used. From the overall performance, the results from heuristic rules achieved accuracy with F-scores of 85.87%, 74.34% and 88.09% for AMOD, ADVMOD and DOBJ, respectively. Conversely, the results from CSR (Hu and Liu, 2006) achieved accuracies of 85.37%, 83.10% and 81.45% for AMOD, ADVMOD and DOBJ, respectively.

#### **2.1.1.4 Word-level sentiment analysis**

Word level sentiment analysis is commonly used to classify contents (word) from document and sentence levels. There are two methods that can be used: lexicon-based and corpus-based (Taboada *et al.*, 2009; Wan, 2009; Petz *et al.*, 2012).

##### **I. Lexicon-based methods**

Lexicon-based methods can be referred to as dictionary-based methods. This method uses the degree of sentiment to measure the polarity derived from text (Wan, 2009). For example, a ‘good’ positive score is 0.75, a negative score is 0 and a neutral score is 0.25 (Baccianella *et al.*, 2010a). In general, a lexicon refers to the collection of information about the words of a language, including the lexical categories to which they belong.

Kim and Hovy (2004) determined the sentiment of opinion under the topics of illegal alien, term limits, gun control and NAFTA. One hundred pieces of data were collected from a Document Understanding Conference (DUC) 2001 corpus<sup>15</sup>. Two human annotators were used to classify the data into three polarities: positive, negative and neutral. Their agreement was measured using Siegel and Castellan’ Kappa (Siegel and Castellan, 1988), which is a statistical method for measuring the agreement, that has striking similarities to

---

<sup>15</sup> [http://www-nlpir.nist.gov/projects/duc/data/2001\\_data.html](http://www-nlpir.nist.gov/projects/duc/data/2001_data.html)

Krippendorff's alpha (Krippendorff, 2011) (Di Eugenio and Glass, 2004). The Siegel and Castellan Kappa value showed 0.91, which is considered both acceptable and reliable.

The contexts defined as an explicit or implicit expression in the text of the author were identified as positive, negative or neutral with regards to the topic. To avoid the problem of differentiating between shades of sentiments, the problem is simplified to identify only expressions of positive, negative and neutral sentiments, together with their holders. However, the sentences in which some sentiment exists but do not express any sentiment will be returned in a separate set.

There are four steps in the experiment. Firstly, the sentences that contain both a topic phrase and holder candidates were selected. Secondly, the holders were delimited based on regions of opinion. Then, the sentences identified potential holders of an opinion by using tagging processes to tag a person's name, company's name and gender, called 'named entity tagger'.

A tool used for a named entity tagger that Kim and Hovy (2004) chose is BBN Identifinder (Bikel *et al.*, 1999). Kim and Hovy (2004) identified the sentiments region by using a scope near each holder and any sentiments that sat outside the region were ignored. Finally, the sentences were split into words to classify the words' polarity using WordNet (Fellbaum, 2010; Princeton University, 2010) (Section 2.1.1.2). However, in the testing system, Kim and Hovy (2004) mentioned that the holders annotated by humans were run first and followed the same models as the automatic holder finding strategies. The results revealed that the accuracy performance achieved 81% with a manually provided holder while the automatic holder detection achieved 67%.

Wu *et al.* (2009) integrated the sentiment orientation of the documents by extending the algorithm that was used to implement the rank sentences, called a graph-ranking algorithm for sentiment transfer. Sentiment transfer is a field in natural language processing which is generally separated into two groups: those that need a small number of labelled training data and those that do not need labelled data for the new domain (Aue and Gamon, 2005; Wu *et al.*, 2009). The datasets were collected from online reviews in the Chinese language by focusing

on three topics: electronics, stock and hotel reviews. They were manually assigned labels as positive and negative. The polarity scores were assigned to a list of words to classify opinions based on the given topic and a set of related texts.

During the experiment, a prototype classification algorithm from Tan *et al.* (2005) and default setting of LibSVM (Chang and Lin, 2011) were used as the baselines. LibSVM (Chang and Lin, 2011) is an open source library for SVM (Kecman, 2005) (Section 2.2). Moreover, both prototype classification algorithms from Tan *et al.* (2005) and LibSVM (Chang and Lin, 2011) were combined with Wu *et al.* (2009)'s algorithm that extended the graph-ranking algorithm for sentiment transfer, which Wu *et al.* (2009) named as OurApproach. Moreover, Wu *et al.* (2009) also compared those results with Structural Correspondence Learning (SCL) (Blitzer *et al.*, 2007). SCL (Blitzer *et al.*, 2007) is a sentiment transfer algorithm that automatically induces correspondence among features from different domains (Wu *et al.*, 2009). The overall average results revealed that the combination of a prototype classification algorithm and OurApproach achieved better results than the others with a 78.70% accuracy.

We are not sure which classifier is meant to be the prototype classification algorithm by Wu *et al.* (2009). Tan *et al.* (2005) used two base classifiers which are centroid classifiers (Hanand and Karypis, 2000) and the Naïve Bayes (Tan *et al.*, 2009) classifier (Section 2.2). Centroid classifiers (Hanand and Karypis, 2000) is an algorithm that provides a simple and efficient method for automatic document classification (Tan *et al.*, 2005). Document classification is a task of machine learning for grouping documents into categories based upon their contents.

## **II. Corpus-based methods**

Corpus-based methods concerned train classifiers by using a corpus of documents that are labelled with polarity (Wan, 2009). In general, corpus (plural corpora) refers to a large collection of text that is used in NLP.

For example, McDonald *et al.* (2007) predicted sentiments at sentence level and document level by using Conditional Random Fields (CRFs) (Lafferty *et al.*,

2001). CRFs (Lafferty *et al.*, 2001) is a structured model that defines the probability over the labels conditioned on the input using the property that the joint probability distribution over the labels factors over clique potentials in undirected graphical models (Lafferty *et al.*, 2001). The datasets were collected from product reviews, after removal of duplicate reviews and reviews that had insufficient text or were spam based on three topics: car seats for children, fitness equipment and Mp3 players.

The documents were annotated by humans, whether they had positive or negative polarity. Next, the documents were split into sentences and annotated by a single annotator using positive, neutral and negative polarity. However, punctuation was also used for making decisions around sentences' polarity; for example, exclamation points, smiley/frowny faces, question marks. Therefore, the system consisted of three baselines: a document classifier, sentence classifier and sentence structure. The document classifier was used to predict only document labels. The sentence classifier was used to predict sentence labels in isolation; in other words, without consideration for either the document or neighbouring sentence sentiment. The sentence structure classifier was similar to the sentence classifier, but this classifier used a sequential chain model to learn and classify sentences.

The models of McDonald *et al.* (2007) used 10-fold cross-validation (Section 2.1.1.3) and trained using a Margin Infused Relaxed Algorithm (MIRA) (Crammer and Singer, 2003; McDonald *et al.*, 2005) learning algorithm. MIRA (Crammer and Singer, 2003; McDonald *et al.*, 2005) is an online machine learning algorithm that relies only on inference to learn the weight vector. The findings revealed a significant increase in performance when labelling decisions between sentences is modelled. Conversely, document-level performance can be improved by incorporating sentence-level decisions. However, the results show accuracy performance at 62.6% and 82.8% for sentence and document levels, respectively.

In, the previous related work, the polarity of sentiment did not have to be 'positive' and 'negative'. There can be more than two labels of polarity (Read, 2009). For example, Mihalcea and Liu (2006) used a corpus-based approach to

classify blogposts that were collected from LiveJournal. A corpus-based approach is a method that trains sentiment classification by using a corpus of documents that are labelled with polarity (Wan, 2009). Corpora are large sets of texts. Mihalcea and Liu (2006) annotated datasets manually using ‘happy’ or ‘sad’.

In the data pre-processing, the blogposts that contained 100 – 8,000 characters and Standard Generalised Markup Language (SGML) (International Organization for Standardization, 1986) tags were removed. SGML (International Organization for Standardization, 1986) tags were used in the document such as Hypertext Markup Language (HTML) tags, which are used on websites. The N-gram<sup>16</sup> feature is a sequence of n consecutive words of size n, and was used in the experiment without using any additional lexicons. The experiments were divided into two tasks. The first was to classify the dataset using ‘happy’ and ‘sad’ labels with a unigram feature. In the second task, the words in the datasets were identified as the factor related to happy and sad and tested using the bigram and trigram features. The results achieved of 79.13% accuracy for the first task, while the second task achieved a slightly lower accuracy of 77.24% and 76.50% for the bigram and trigram, respectively.

Pak *et al.* (2012) used both a machine learning algorithm and manually-defined transducers to detect sentiment in suicide notes. The datasets comprised of 900 notes (Pestian *et al.*, 2012). 15 categories were used to identify the opinion expressed in the notes: instructions, information, hopelessness, guilt, blame, anger, sorrow, fear, abuse, love, thankfulness, hopefulness, happiness/peacefulness, pride and forgiveness. As mentioned above, there are two approaches in this experiment: the machine learning algorithm and manually-defined transducers. The default setting of LIBLINEAR (Fan *et al.*, 2008) was used in the machine learning based approach. LIBLINEAR (Fan *et al.*, 2008) is a package providing a library for linear classification. Linear classification is a learning technique that is used for large sparse datasets with a number of instances and features (Fan *et al.*, 2008).

Pak *et al.* (2012) used six features to build the classification model. The first and second features were n-grams and a dictionary from General Inquirer (GI)

---

<sup>16</sup> Unigram is a collection of text in size one while bigrams and trigrams are a collection of text in size 2 and 3, respectively.



(Stone *et al.*, 1968) (Section 2.1.1.2). The third feature was part-of-speech tagging using TreeTagger (Schmid, 1995). TreeTagger (Schmid, 1995) is an annotation tool which labels each word with part-of-speech and lemma (Schmid, 1995). The fourth feature was the lexicon which is used from the Affective Norms of English Words (ANEW) (Bradley and Lang, 1999). ANEW (Bradley and Lang, 1999) is a lexicon that was developed from a large number of English words to provide a set of normative emotional ratings. The fifth feature is the dependency graph. Dependency graph is the name of the process for extracting the sentence dependency in which the process from the Stanford lexical parser (The Stanford Natural Language Processing Group, 2002) (Section 2.1.1.2) is used to create patterns of sentiment expression. The final features were heuristic features which were used for adding the position of the sentence with respect to the beginning of the note and the presence of the following keywords in the sentence such as god, thank, please, car and Cincinnati<sup>17</sup>.

For the approach of manually-defined transducers, Pak *et al.* (2012) identified emotions in the notes using extraction patterns. Extraction patterns are methods for extracting a pattern from sentences. Pak *et al.* (2012) decided to define the pattern manually using a limited number of training data and all target classes. These patterns combined three features. The first feature was part-of-speech tagging. The second feature was a surface-level token which was used to extract the original word from the token. The final feature was Lemmas. Lemmas are words which stand at the head of a definition in a dictionary. For example, write, wrote, written are forms of the same units of meaning, but write is the lemma. After the pattern process, Pak *et al.* (2012) detected texts by using finite-state transducers. Finite-state transducers are used to automatically tag pattern occurrences in the input text. All the cascaded transducers were applied in the final classification, one after the other in a specific order to avoid possible problems from the expression which may be identified by several transducers. After that, both approaches were combined and achieved better accuracy performance than using each one of them alone, with an F-score of 53.83%.

---

<sup>17</sup> Cincinnati is the name of an industrial city in Ohio.

Keshtkar and Inkpen (2010) used a corpus-based method to extract four collections of datasets: LiveJournal dataset (Mishne, 2005), a text affect dataset (Strapparava and Mihalcea, 2007), fairy tales dataset (Alm *et al.*, 2005) and annotated blog dataset (Aman and Szpakowicz, 2007). The extension of WordNet (Fellbaum, 2010; Princeton University, 2010) (Section 2.1.1.2) that has information about the emotions, as a set of seed words for helping to label the datasets, called WordNet Affect (Strapparava and Valitutti, 2004), was used to assign the labels of six classes to the dataset. The classes are ‘happiness’, ‘sadness’, ‘anger’, ‘disgust’, ‘surprise’, and ‘fear’. The datasets were trained and classified by using ensemble learning methods called bagging (Sun and Pfahringer, 2011) (Section 2.3.1). The results showed that Keshtkar and Inkpen (2010) achieved an F-score of 87.30%, 85.33%, 86.63%, 86.22%, 85.76% and 84.36% for the classes of disgust, fear, anger, happiness, sadness and surprise, respectively.

### **2.1.2 Purposes of sentiment analysis**

The purpose of sentiment analysis is to identify opinions or attitudes in terms of polarity. The polarity is the perspective of the person. Sentiment analysis has been used in many ways such as advertising (Jin *et al.*, 2007) and marketing and production (Mishne and Glance, 2006). In terms of advertising, the internet is the best medium through which to promote businesses as it will reach target group of customers in which sentiment analysis could be used to help ensure that the website’s contents fit with the commercial content so that it is not detrimental to the reputation and popularity of the company and/or brand (Jin *et al.*, 2007). Figure 2.1 displays a page from Yahoo in which users searched for the keyword “Samsung Galaxy”, and the page extracted showed some relevant advertisements on the right side of the page (in the red rectangular box).

For example, Jin *et al.* (2007) classified webpages to detect whether a publisher’s webpage contains sensitive content and is appropriate for showing advertisements on it. Sensitive content taxonomy was designed, although Jin *et al.* (2007) did not explain the design clearly due to a certain policy for their company. Jin *et al.* (2007) mentioned that the taxonomy is flexible and can be trained using

different classifiers with different granularity such as category level and sub-category level (leaf level). Each leaf level is tagged as sensitive or non-sensitive which leads to building a simple binary classifier for judging the webpage. The next step was to collect and classify the webpages. The data was collected from labelled and unlabelled webpages. These were split into the phrase (key term) with the term and then, Jin *et al.* (2007)'s keyword suggestion tools were applied to the terms to get an expanded set of related terms. Jin *et al.* (2007) did not mention any details about the keyword suggestion tool. For example, 'sex education' is a category and 'safe sex' and 'teen sex education' are related terms.

After this, the data was passed through feature processing. This extracted useful text information such as title, data and body; identifies phrases, bigram and trigram from the extracted text; remove stopwords; finds document length and text patterns. Text patterns (König and Brill, 2006) are defined as an ordered sequence of words which is similar to the notion of the regular expression in the Perl language (Jin *et al.*, 2007). A regular expression is a method that defines the pattern of the content. Jin *et al.* (2007) adopted SVM (Kecman, 2005) (Section 2.2) and Logistic Regression (Kleinbaum and Klein, 2010) (which is a statistical method for determining the data when there are binary variables), to use with binary classifiers and hierarchical classifiers. A hierarchical classifier is a classifier that ignores the hierarchical structure of the taxonomy but only classifies pages directly into one of the leaf nodes (categories) language (Jin *et al.*, 2007).

In the experiment, the labelled pages were used to build the initial classifier. Then, unlabelled pages were applied with this classifier in which each unlabelled page was assigned a class label along with a probability value. In hierarchical classifier methods, there are two methods. Firstly, unlabelled pages that have high probabilities were assigned to the category as training pages for the next iteration. Secondly, it is necessary to request labels of a set of unlabelled pages which might provide more complementary information to the current classifier. The accuracy achieved was 59% and 55% from SVM (Kecman, 2005) (Section 2.2) and Logistic Regression (Kleinbaum and Klein, 2010), respectively. In the binary classifier methods, the hierarchical multiclass classifier was run first for getting the leaf category of the input (webpages) to predict whether or not the pages

contain sensitive or non-sensitive contents. When comparing these approaches, the results from the binary classifier achieved a better accuracy than the hierarchical classifier at 81% and 76% from SVM (Kecman, 2005) (Section 2.2) and Logistic Regression (Kleinbaum and Klein, 2010).

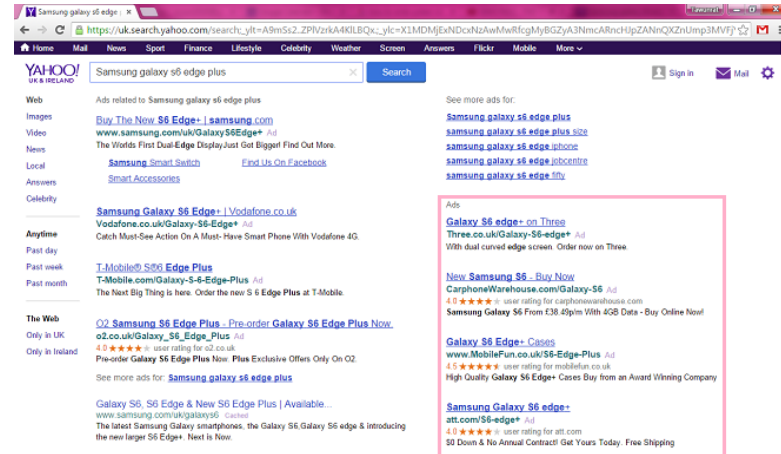


Figure 2-1: Example of webpage with advertising's contents

Advertising, marketing and production are the main keys for a company and a brand can use sentiment analysis to predict price and demand for the products. For example, Mishne and Glance (2006) used sentiments that bloggers expressed about a movie for predicting the sales. The separate periods of the weblog were used: before and after the movie's release. The data was collected from the Internet Movie Database (IMDB) by using a selection of 49 movies that were released between February and August 2005. Posts that were related to the movie were selected using keywords to extract the contexts around the hyperlinks to the movie's IMDB page or around exact matches of the movie names. The number of keywords varied from six words to 250 words. The information on the overall contexts are focused on date, sales, screen numbers of the opening weekend; income per screen; pre-release and post-release data such as references in weblogs, context length, positive and negative references. Once the contexts were extracted, the methods from Nigam and Hurst (2004) were adopted to calculate the sentiment values of the contexts. In Nigam and Hurst (2004)'s process, the input was tokenised and tagged using part-of-speech information. Next, semantic tagging was used to add polarity to each token (positive or negative) based on

Nigam and Hurst (2004)'s lexicon. Simple linear part-of-speech tag patterns were applied to form chunks (noun, adjective, adverb and verb). A chunk is a process that is used for moving a group of information. 'Chunk up' refers to moving to more general or abstract pieces of information. 'Chunk down' means moving to more specific or detailed information. The chunked input is then further processed to form a high-order grouping of a limited set of syntactic patterns. These patterns were designed to cover expressions that associated polarity with some contents and those expressions that toggle the logical orientation of polar phrases.

After assigning polarity to the movie, Mishne and Glance (2006) measured the relationship between several sentiment-derived metrics and both income per screen and raw sales by using the Pearson correlation. The Pearson correlation is the common correlation measurement that is used with two variables that are normally distributed (Howitt and Cramer, 2003; Rumsey, 2007; Downey, 2014). Correlation measurement is a technique that is used to measure the relationship between two or continuous variables. Besides the sentiment-related correlation, Mishne and Glance (2006) also measured the correlation of the number of posts that referred to the movies. However, the measurement was done separately for both pre-release and post-release contexts. Mishne and Glance (2006) reported that in over half of 49 movies, there was a good correlation between pre-release positive sentiment and sales. Mishne and Glance (2006) concluded that their results indicated that sentiment could be effectively used in predictive models for sales in conjunction with movie genre and season. Nevertheless, Mishne and Glance (2006) did not clearly mention the results of sentiment analysis that were achieved but instead mostly focused on results from the correlation process.

Besides the marketing and production, sentiment analysis can be used to analyse product reviews from customers (Grabner *et al.*, 2012). For example, Grabner *et al.* (2012) used sentiment analysis to classify customers' reviews of hotels by using a star rating to categorise the reviews as bad, neutral and good, for further details see Section 2.1.3.5.

Moreover, political organisations (Tumasjan *et al.*, 2010) also used sentiment analysis to analyse public opinion about policies, legislation, politics, government agencies, etc. Twitter has been analysed by researchers for political

postings on microblogs. For example, Tumasjan *et al.* (2010) investigated whether or not Twitter can be used to reflect the political sentiment and predict the election. The data was collected from tweets that contained the names of the six parties of the German Parliament from August to September 2009: CDU, CSU, SPD, FDP, LINKE and Grune.

The tweets were then translated into English. Tumasjan *et al.* (2010) computed the degree of sentiment of tweets using Linguistic Inquiry and Word Count (LIWC) (Pennebaker *et al.*, 2001). LIWC (Pennebaker *et al.*, 2001) is a text analysis software that was developed to estimate the emotional, cognitive and structural components of text (Tumasjan *et al.*, 2010). LIWC (Pennebaker *et al.*, 2001) is used for the analysis of text files on a word-by-word basis using an internal dictionary of 6,400 of the most common words and word stems. LIWC (Pennebaker *et al.*, 2001) work by matching each word in a text to a word in the dictionary and the associated word characteristics are extracted.

In order to analyse the political sentiment of the tweets, Tumasjan *et al.* (2010) generated multi-dimensional profiles of the politicians in the sample using the relative frequencies of LIWC (Pennebaker *et al.*, 2001) category word counts. Tumasjan *et al.* (2010) collected the dataset of Tweets by using the keywords of 6 parties represented in the German parliament. The results showed that, the positive emotions clearly outweighed the negative emotions. There are two aspects that were used to investigate whether Twitter could be used to predict the election results. The first aspect was to compare the share of attention the political parties received in tweets with the election results. The second aspect is to analyse whether tweets can inform about the ideological ties between parties and potential political coalitions after the election. Tumasjan *et al.* (2010) showed that the relative volume of tweets mirrors the results of the federal election closely in which the MAE (Mean Absolute Error) of the number of tweets to be a predictor of the election results is 1.65%.

Tumasjan *et al.* (2010) found Tweets are used for the discussion of political opinions. Tumasjan *et al.* (2010) claimed that the number of Tweets that mention political parties can plausibly reflect the election results. Following, Choy *et al.* (2011) who achieved 6.59 % and 5.15% between predictions from Tweets and the

actual value of Singapore election for Toony Tan and Tan Cheng Bock, respectively. Also, Gaurav *et al.* (2013) achieved an average Root Mean Square Error (RMSE)<sup>18</sup> less than 0.03 when computed a Moving Average Aggregate Probability (MAPP)<sup>19</sup> of a candidate over a period of 7 days. Both of Choy *et al.* (2011) and Gaurav *et al.* (2013) used candidate names data collection.

In fact, there is research from Jungherr *et al.* (2012) who replicated Tumasjan *et al.* (2010) by adding a seventh party, called the Pirate Party (Piraten). The Piraten was supported<sup>20</sup> in online forums, blogs and on Twitter and was mentioned in Tweets more than any other parties at 34.8%. However, the election results showed that Piraten gained only 2.1% of the votes. Hence, Jungherr *et al.* (2012) conclude that, Twitter is not an accurate election predictor.

Therefore from both groups, it may be concluded that, twitter is not an accurate election predictor or at least a controversial election predictor.

Researchers in the field of psychology (Hobson *et al.*, 1998; Domhoff, 2003; St-Onge *et al.*, 2005) are also concerned with emotion, which plays an important role in dreams (Hobson *et al.*, 1998; Domhoff, 2003; St-Onge *et al.*, 2005). Normally, the emotions in dreams are assessed and analysed by the dreamers. For example, Nadeau *et al.* (2006) analysed sentiments in dreams using Naïve Bayes (Tan *et al.*, 2009) (Section 2.2) and Linear Regression (Kleinbaum and Klein, 2010). Linear Regression (Kleinbaum and Klein, 2010) is a classifier method that is used to determine the initial position of two classes. The data collected comprised of 100 dreams. The dreams were from the dream bank that was created for the normative study of a dream. Each dream was collected by asking volunteers to write down the dreams they remembered over a three week period. Two annotators were asked to annotate the contents by using a scale from zero (positive, neutral) to three (highly negative). Their agreement is measured by using an inter-judge agreement which is the percentage of incidents that both judges decide on the same contents, and mean squared error (MSE) (Koga *et al.*, 1981). The mean squared error (MSE) (Koga *et al.*, 1981) is used to calculate the

---

<sup>18</sup> Root Mean Square Error (RMSE) is used for measuring the differences between the estimated probabilities and actual outcome.

<sup>19</sup> MAPP is the approaches that compute the probability of a candidate winning per day and then use the mean of probability in a week.

<sup>20</sup> This supported is led to widespread in German media and academia if online channels would change political participation in German.

average of the differences between each predicted value and its corresponding correct values (Patil *et al.*, 2010; Witten *et al.*, 2011). The positive scale is at zero due to their agreement with the positive scale, which is low at 57.7%, MSE 0.54, while the negative is 80.8%, MSE 0.19. An inter-judge agreement is a method used to calculate the percentage of agreement of both judges (annotator) (Hayes, 2008).

After that, in the analysis process, four strategies were used. The first and the second strategies are General Inquirer (GI) (Stone *et al.*, 1968) (Section 2.1.1.2) and LIWC (Pennebaker *et al.*, 2001). The third strategy is the bag-of-words. The final strategy is weighted GI and HM lexicons (Turney and Littman, 2003). The weighted GI and HM lexicons (Turney and Littman, 2003) were produced in a process which weights were assigned to the lexicons to represent their orientation and strength of the words in it (Turney and Littman, 2003). The outputs were generated using WEKA (Hall *et al.*, 2009) (Section 2.2.1). The results achieved the best accuracy at 50%, MSE 0.577 from using Liner Regression with GI. The results from Liner Regression with LIWC achieved an accuracy of 48% and MSE 0.608, which is better than the accuracy from Naïve Bayes with the bag-of-words and Liner Regression with weighted GI and HM. They achieved accuracies of 38%, MSE 1.392 and 35% and MSE 0.865, respectively.

For the purposes of sentiment analysis, there is a question, how can the degrees of sentiment in contexts be measured? The answers are presented in the following section.

### **2.1.3 Processes to measure degree of sentiment**

According to the previous question, there are typically seven criteria that could be used such as the polarity of words and range of polarity, human classification, emoticons, linguistic features, sentiment resources and sentiment lexicons. The details of each of these are explained in the following sections.



### 2.1.3.1 Sentiment analysis via polarity of words

Words can be used to assign and label the polarity, such as positive, neutral and negative. For example, Wilson *et al.* (2005b) used phrase level sentiment analysis (Section 2.1.1.3) to analyse only the dataset from the MPQA corpus (Wiebe *et al.*, 2005). The MPQA corpus (Wiebe *et al.*, 2005) is a corpus that contains news articles. Wilson *et al.* (2005) annotated the dataset from the MPQA corpus (Wiebe *et al.*, 2005) manually to use in this experiment which had two tasks. The first task was to classify whether the phrases should be positive, neutral or negative. The second task was rather similar to the first task, but adds the label called, ‘both’ to the phrases that have both positive and negative labels. The results showed that, Wilson *et al.* (2005b) achieved high accuracy from the first task at 75.90% while the second task achieved 65.70%. From the results, it can be said that adding more labels did not always improve the accuracy performance.

Agarwal *et al.* (2009) used phrases from the MPQA corpus (Wiebe *et al.*, 2005) in the experiment. Before classifying the label of each sentence, the sentences were classified as subjective or objective (neutral) and after that, the subjective sentences were assigned labels such as being positive or negative. Each phrase was assigned scores by using the Dictionary of Affect in Language (DAL) (Whissell, 2009). DAL (Whissell, 2009) is a resource that is used to label the emotion of text. The datasets were divided for use in two tasks: balanced and unbalanced. Balanced meant that the number of phrases which were positive, neutral and negative were equal while unbalanced did not. Each dataset was tested using tripolarity (positive, neutral and negative) and bipolarity (positive and negative) in a 10-fold cross validation (Section 2.1.1.3). Three features were used in the experiment: part-of-speech, n-gram (unigram, bigrams, and trigrams) and chunks (section 2.1.2). The results showed that using all features with bipolarity achieved high accuracy with the balanced and unbalanced task at 82.32% and 84.08%, respectively. However, the results of the combination of each feature were not shown clearly.

Besides the positive, neutral and negative labels, there are still others that were used depending on the decision of researchers (Read, 2009), as mentioned in Section 2.1.1.4. Further details about the number and range of polarities are

presented in Section 2.1.3.5. Furthermore, human annotators are also used to assign the polarity of contents. Further details of human annotators are described in the following section.

### **2.1.3.2 Sentiment analysis via human classification**

When using humans to classify the contents, the researchers should find more than two annotators to score the words using various ranges. The ranges could vary, depending on the agreement between the researchers and annotators. After that, a statistical measure of the agreement of annotators will be used.

For example, Devitt and Ahmad (2007) examined the relationship between financial markets and news, in particular the polarity of financial news. This data was collected from the national media and international news about the bidding for Ryanair in 2006. A set of 30 texts from the data was chosen for use as a gold standard. Gold standard refers to the data that is labelled by human annotators as having the correct polarity, which will be used to measure the accuracy of the machine process.

Devitt and Ahmad (2007) selected three human annotators for annotating the data by ranking from 1 (very negative) to 7 (very positive). There were three elements that annotators had to work on. The first was to annotate the text. The second was to rate the semantic orientation of the texts with respect to the bidding. There were two players in the bidding war, Ryanair and Aer Lingus. Finally, the third step was to rate their personal attitudes towards those airlines. All three steps used the same ranking in the annotations. Once all annotations were received, the statistical method, Krippendorff's alpha (Krippendorff, 2011) (Section 2.1.1.4) was used to measure their agreement. For the agreement on the general ranking scale, the Kappa value was 0.1685 which represents little agreement. On the other hand, the agreements on the polarity rating of those two airlines gave Kappa values of 0.5795 and 0.5589, respectively. These values show on acceptable degree of agreement. To classify the polarity of the other text, SentiWordNet (Baccianella *et al.*, 2010a) (Section 2.1.3.6) and WordNet (Fellbaum, 2010; Princeton University, 2010) (Section 2.1.1.2) were used. The overall accuracy performance achieved an F-score of 46.67%. The following

section contains details of emoticons used to assign sentiment labels to the contexts.

### **2.1.3.3 Sentiment analysis via emoticons**

Icons that can be used to express emotion are called ‘Emoticons’ (Witmer and Katzman, 1997; Danet *et al.*, 1997; Derks *et al.*, 2008; Wang *et al.*, 2014). These are usually used in social media and short messages, such as Facebook, Twitter and SMS.

For example, Aisopos *et al.* (2011) analysed sentiments within Twitter. The data was collected from real-time Tweets, although Aisopos *et al.* (2011) randomly selected 1 million tweets for each polarity: positive, neutral and negative. The data that lacked any polarity indicator or contained both positive and negative emoticons was assigned as neutral. The data contained positive emoticons: assigned as positive polarity: ':)', '(:', ':-)', '(-:', ':)', '(:', ':D' or '=)'. In contrast, the data contained negative emoticons: assigned as negative polarity: ':(', '):', ':-(', ')-:', ': (' or ':)'

Two machine learning algorithms from WEKA (Hall *et al.*, 2009) (Section 2.2.1) were used: Naïve Bayes Multinomial (NBM) (McCallum and Nigam, 1998a; Bermejo *et al.*, 2011) which is a modification of Naïve Bayes (Tan *et al.*, 2009) and the C4.5 Decision Tree algorithm (Quinlan, 1993; Polat and Gunes, 2009) which is used to generate a decision tree. A decision tree uses a tree graph or model to make the decision (Safavian and Landgrebe, 1991). Aisopos *et al.* (2011) also used the N-gram Graph Based method from JInsect (Giannakopoulos and Karkaletsis, 2009) in the experiment. The N-gram Graph Based method is a document representation model that improves the character n-grams model by adding contextual information instead of generating a plain bag of n-grams (Aisopos *et al.*, 2011). JInsect (Giannakopoulos and Karkaletsis, 2009) is an open source and JAVA-based toolkit for the N-gram Graph Based method.

After using the N-gram Graph Based methods, the results showed that the C4.5 Decision Tree algorithm (Quinlan, 1993; Polat and Gunes, 2009) achieved higher accuracy performance than NBM (McCallum and Nigam, 1998a) at similarity 66.77% and discretizing at 65.34%. in addition to emoticons, the

feature-based method can be used to classify sentiment, as described in the following section.

#### **2.1.3.4 Sentiment analysis via feature-based analysis**

The feature-based analysis is focused on the target entities and components (attributes and features) of the opinions. Such targets could be the service, product, organisation and topic. For example, Hu and Liu (2004) analysed and summarised customer product reviews by using a feature-based approach and focusing on the product on which the customers have expressed their opinion (positive or negative). The online customers' reviews were collected based on five products. They were two digital cameras, a DVD player, an Mp3 player and a mobile phone. The first 100 reviews of each product were selected. Then, the data generated a part-of-speech tag using the NLPProcessor linguistic parser (Infogistics Ltd., 2000) which is an online programme used to parse and produce part-of-speech tags for each word. The noun and noun phrases were assigned as product features and adjectives were used as opinion words. If the features were frequently mentioned by the customer, the features were counted as frequent features.

Next, WordNet (Fellbaum, 2010; Princeton University, 2010) (Section 2.1.1.2) was used to predict the subjective semantic orientations. In order to predict the orientation of the opinion sentences, the dominant orientations of the opinion words in the sentences were used to determine the orientation of the sentences. If the sentiment opinion prevails, the opinion sentences were considered to be either positive or negative. Conversely, if the sentences contained the same number of positive and negative opinion words, the orientation of the previous opinion sentences was used to make predictions. Two steps were used to generate the final feature-based review summary. Firstly, for each discovered feature, related opinion sentences were put into positive and negative categories according to the opinion sentences' orientation. A count was computed to show how many reviews gave a positive or negative opinion of the feature. Finally, the frequency of the features that appear in the reviews was used for ranking. The overall average sentence orientation accuracy achieved was 84.20%. However, Hu and Liu (2004) did not declare the features on which the

products were focused on. The following section describes the polarity range used.

#### **2.1.3.5 Sentiment analysis via range of polarity**

The range of polarity uses a set of numbers as the labels. There are researchers who have used the range of polarity such as Grabner *et al.* (2012), who collected customer reviews from TripAdvisor by focusing on travel and vacation services. There were two approaches used in the experiment. The first approach was to classify the reviews into five star categories while the second approach was to classify the reviews into three categories: good, neutral or bad. For the first approach, the values of each star has been assigned using a weight of -2 for 1 star, a weight of -1 for 2 stars, a weight of 0 for 3 stars, a weight of 1 for 4 stars and a weight of 2 for 5 stars. On the other hand, a weight of -2 for bad, a weight of 0 for neutral and a weight of 2 for good was assigned in the second approach.

For the classification process, Grabner *et al.* (2012) did not use any machine learning algorithms but instead the method from Pang and Lee (2008) was used. The method (Pang and Lee, 2008) was used to summarise the polarity of each word in the document to perform the polarity sentiment of the document. The results showed that Grabner *et al.* (2012) achieved better accuracy by using the classes of good, neutral and bad than using the range of 5 stars with an average F-score of 54.00% and 35.40%, respectively. The results showed that fewer class labels achieved better performance than many class labels. The Grabner *et al.* (2012) experiment was not compared against any machine learning algorithm so it is not clear which achieved better between the base learning and machine learning algorithms. Other than using a range of polarity from the reviews, there are sentiment resources that used a range of a set of numbers to label the sentiment of words. The details of these sentiment resources are described in the following section.

#### **2.1.3.6 Sentiment analysis via sentiment resources**

Sentiment resources are resources that automatically extract sentiment from phrases or sentences. Sentiment resources are composed of a word list of

sentiment terms and opinion lexicons which are mostly used in English. There are two sentiment resources that will be described in this section which are SentiStrength (Thelwall *et al.*, 2010b) and SentiWordNet (Baccianella *et al.*, 2010a).

SentiStrength (Thelwall *et al.*, 2010b) is a sentiment analysis methodology used to judge whether a sentence has a positive or negative sentiment. SentiStrength (Thelwall *et al.*, 2010b) was developed using nearly 4,000 comments on MySpace. MySpace<sup>21</sup> is a social network website, which has services for creating a blog, saving pictures, music and video and enables users to connect to the others. Thelwall *et al.* (2010a) used three annotators. Their agreements were measured using the static method, Krippendorff's alpha method (Krippendorff, 2011) (Section 2.1.1.4). The data was separated into two groups: trial data and testing data. Trial data was used to identify algorithms for judgment and suitable scales. Algorithms were identified using a range of 1 to 5. Thelwall *et al.* (2010a) were used alongside testing data for the final judgment and these will be a lexicon of SentiStrength (Thelwall *et al.*, 2010b). SentiStrength (Thelwall *et al.*, 2010b) is available to use free of charge and has been used by several researchers.

For example, Pfitzner *et al.* (2012) used SentiStrength (Thelwall *et al.*, 2010b) to assign the scores of the dataset. The score of each dataset that was annotated by SentiStrength (Thelwall *et al.*, 2010b) was converted to -1 if the negative score was more than the positive score, 0 if the scores were equal and 1 if the positive score was more than the negative score. Preethi *et al.* (2012) investigated online hotspot forums called, forums.digitalpoint.com using SentiStrength (Thelwall *et al.*, 2010b) to assign sentiment scores of the existing text in forums that were concerned with Search Marketing, Publisher Network and General Marketing, directly.

SentiWordNet (Baccianella *et al.*, 2010a) is the result of automatic annotation of all the synsets of WordNet (Fellbaum, 2010; Princeton University, 2010) (Section 2.1.1.2), according to the labels of positive, negative and neutrality, to which each synset was allocated three numerical scores Pos(s),

---

<sup>21</sup> <https://myspace.com/>

Neg(s) and Obj(s). Each of the three scores ranged from 0.0 to 1.0 and the summary is 1.0 for each synset. This means that, there is the possibility of having non-zero scores for all three. The methods used to generate SentiWordNet (Baccianella *et al.*, 2010a) were adapted from the methods of PN-polarity and SO-polarity (Esuli and Sebastiani, 2006b). PN-polarity is used to determine whether the opinion is positive or negative, while SO-polarity determines whether the opinion is subjective or objective. The methods rely on the quantitative analysis of annotations associated with synsets and on the use of the resulting quantity term representations for semi-supervised synset classification (Esuli and Sebastiani, 2007). Semi-supervised classification (Zhu *et al.*, 2009a) is a machine-learning technique for use with both labelled and unlabelled data. SentiWordNet (Baccianella *et al.*, 2010a) is a freely available and widely used electronic resource.

For example, Denecke (2008) used SentiWordNet (Baccianella *et al.*, 2010a) in the task of sentiment analysis for multilingual use. Denecke (2008) annotated the datasets using the combination of each subset of each polarity and divided by the number of the subsets. After that, the score of the sentences were summarised using the scores from each polarity of words and dividing them by the number of words in the sentence that were considered. Denecke (2008) achieved an accuracy of 66% for the summarisation of sense in SentiWordNet (Baccianella *et al.*, 2010a).

Ghorbel and Jacot (2011) used a lexicon-based method of sentiment analysis to analyse French movie reviews. Lexicon-based methods can be referred to as dictionary-based methods that use the lexicon to measure the polarity of text. Lexicons are a set of words used to express emotions and opinions in sentiment contexts. There are five features that have been used: unigram, part-of-speech, polarity stopwords and lemmatisation. Stopwords (Bird *et al.*, 2009b) can be defined as words that are frequently used, are less important and do not have meaning such as *a*, *an* and *the*. Lemmatisation is a process that analyses word using vocabulary and returns the base form of a word, which is known as the lemma. The polarity of datasets were annotated using SentiWordNet (Baccianella *et al.*, 2010a) after translating the datasets to the English language according to,

SentiWordNet (Baccianella *et al.*, 2010a), which only works with English language. The standard default setting of SVM (Kecman, 2005) (Section 2.2) from SVMLight (Joachims, 2002a; Joachims, 2002b) were used in the experiment.

Ghorbel and Jacot (2011) achieved an accuracy of 91.50% using just unigrams but after adding the features of lemmatisation and polarity, the accuracy increased to 93.25%. The results revealed that combining features could help to obtain a better performance. Conversely, the accuracy decreased to 92.75% after adding part-of-speech. Part-of-speech affects the results because of the fact that there are a large amount of misspellings in the dataset. Therefore, it can be stated that, part-of-speech was not suitable for the datasets that contained a large amount of misspelling. This could be supported by the work from Go *et al.* (2009). Go *et al.* (2009) classified Twitter using three machine learning algorithms: Naïve Bayes (Tan *et al.*, 2009) (Section 2.2), Maximum Entropy Modelling (Harte, 2011) (Section 2.2) and SVM (Kecman, 2005) (Section 2.2). In the experiment, emoticons have been used as noisy labels in training data to identify the label as positive or negative. Emoticons can be referred to as printable characters of emotion such as :- ) for a smile and :- ( for sad. SVM (Kecman, 2005) used with unigrams obtained a high accuracy of 82.90%. Go *et al.* (2009) stated that using negation and part-of-speech tagging did not help to improve accuracy.

Amiri and Chua (2012) classified the datasets by using sense level polarity based on SentiWordNet (Baccianella *et al.*, 2010a). The datasets from Pang and Lee (2004) and Whitehead and Yaeger (2009) on the topics camera, camp, doctor, music and movie were used in the experiment. The datasets from Pang and Lee (2004) were the collection of movie reviews from Rottentomatoes.com while the dataset from Whitehead and Yaeger (2009) was collected from websites such as Amazon.com, CampRatingz.com and RateMDs.com. SVM (Kecman, 2005) (Section 2.2) was used with three approaches in the experiment. In the first approach, if the positive and negative scores from SentiWordNet were equal, the label would be assigned as positive (SWNOPN).

On the other hand, the others will be labelled as the sentiment of the term by using -1 for negative and 1 for positive (SWNPN) as the second approach.



Finally, the most common sense of each term in SentiWordNet was used to label the term (SWNMCS). The results from the three approaches were compared with the base results. The results showed that, in the topic of the camera, SWNPN achieved accuracy which was more than the base line result at 79.42%. For the topics of camp and music, the use of SWNMCS achieved 80.32% and 72.27%, respectively which was also more than the base results. On the other hand, the use of the three approaches on the topic of doctors and music did not achieve a better score than the base result. Amiri and Chua (2012) did not use the summarisation of sense of each term in the experiment. Therefore there is still no answer, if the summarisation of sense of each term has been used, over whether the performance will achieve better accuracy or not.

After reviewing the related work from Pfitzner *et al.* (2012), Denecke (2008), Ghorbel and Jacot (2011) and Amiri and Chua (2012), used sentiment resources in sentiment analysis. Their work led us to these research questions: (*RQ. 1*) ‘In the comparison of SentiStrength (Thelwall *et al.*, 2010b) and SentiWordNet (Baccianella *et al.*, 2010a), which sentiment resources will achieved better accuracy in the context of data? Moreover, will the accuracy be better than the results from word polarity (positive and negative)?’

Besides these sentiment resources, there are sentiment lexicons that could be used to measure the polarity of text. The details of the sentiment lexicons can be found in the following section.

#### **2.1.3.7 Sentiment analysis via sentiment lexicons.**

Sentiment Lexicons are lexicons (dictionaries) with sentiment values attached to each word. There are sentiment lexicons in the English language such as the Bing Liu Lexicon<sup>22</sup> which is a collection of online customer product reviews (Hu and Liu, 2004), the MPQA Subjective Lexicon<sup>23</sup> is collection of news articles (Wilson *et al.*, 2005b), and the AFINN Lexicon<sup>24</sup> is a list of English words created using the contents from microblogs (Nielsen, 2011a). Moreover, some researchers have used the other sentiment lexicons.

---

<sup>22</sup> <http://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html#lexicon>

<sup>23</sup> [http://mpqa.cs.pitt.edu/lexicons/subj\\_lexicon/](http://mpqa.cs.pitt.edu/lexicons/subj_lexicon/)

<sup>24</sup> [http://www2.imm.dtu.dk/pubdb/views/publication\\_details.php?id=6010](http://www2.imm.dtu.dk/pubdb/views/publication_details.php?id=6010)

For example, Wan (2008) used both Chinese and English lexicons to improve sentiment analysis in the Chinese language by using the combination of lexicons together. The Chinese lexicons were used from a Chinese Vocabulary for sentiment analysis called HowNet-VSA (Dong and Dong, 2006) while the English lexicons were from the MPQA Opinion Corpus (Wiebe *et al.*, 2005). Three machine translations were used to translate Chinese reviews into the English language: Google Translate, Yahoo Babel Fish and Baseline Translation. Baseline Translation (Wan, 2008) was developed using a Chinese-to-English dictionary while Google Translate and Yahoo Babel Fish are the online translation services.

For the process of classification, six methods of ensemble learning were used. The first method was the average combination which uses the average values of each classification. The second method was the weight average combination which uses the average combination of the weight that was assigned to the individual classification. The third method used was maximum value while the forth method used was minimum value. The average of the maximum and minimum values was the fifth method. The final methods was Majority Voting (Polikar, 2012) (Section 2.3). The results revealed that Wan (2008) obtained an F-score of 85.40%, 86.10%, 82.30%, 84.80%, 84.30% and 82.30% from each method, respectively. In the experiment, Wan (2008) did not mention the combination of training data with sentiment lexicons in both languages therefore, would the accuracy achieved have been higher if the combination of sentiment lexicons with training data were used?

Kieu and Pham (2010) developed a system to analyse product reviews on the topic of laptop and desktops in the Vietnamese language. Kieu and Pham (2010) claimed that there is no public corpus for Vietnamese sentiment analysis. Therefore, Kieu and Pham (2010) decided to create a corpus and assigned the polarity using Callisto (Day *et al.*, 2004). Callisto (Day *et al.*, 2004) is a tool that is used to annotate the data. The datasets were annotated by separating them into two groups: word and sentence. After that, Kieu and Pham (2010) created a rule-based system using GATE<sup>25</sup> (Cunningham *et al.*, 2011) (Section 2.2.1). The rule base was composed of word correction, sentiment word recognition, sentiment

---

<sup>25</sup> <http://gate.ac.uk/>

classification and feature evaluation. The results showed an F-score of 77.83% and 62.84% from using word and sentence levels, respectively. Although there is no public corpus for the Vietnamese language, there are corpora in the English language that could be used. If Kieu and Pham (2010) had translated Vietnamese to English and used English sentiment lexicons, would the accuracy have improved?

After reviewed the related works from Wan (2008) and Kieu and Pham (2010), following Wan (2008) used sentiment lexicons in sentiment analysis. Their work led to the following research the questions: (*RQ. 2*) ‘Are sentiment lexicons essential for the sentiment analysis task?’ (*RQ. 3*) ‘How much accuracy performance will be achieved when using only training data?’ (*RQ. 4*) ‘Will the accuracy improve if combined training data and sentiment lexicons are used?’

Beside the above related work, there are workshops that are related to sentiment analysis. In the following section, brief details of the workshop under the Association for Computational Linguistics (ACL), SemEval (Semantic Evaluation)<sup>26</sup> are described.

#### **2.1.4 Comparative assessment of sentiment analysis**

SemEval is an ongoing workshop that has run from 1998 until now with the purpose of evaluating the semantic system. In 2013, there is a task in the SemEval workshop that is concerned with sentiment analysis in Twitter. This task was chosen for the reason that the datasets will be provided by the organisers and the results from the participators can be compared with the effectiveness of each of the techniques that were provided by the participants. The datasets are composed of training data, testing data and the gold standard. The gold standard refers to the testing data that was labelled with the correct polarity. The gold standard will be used to measure the accuracy of the test.

The datasets were annotated using the service provided by Amazon Mechanical Turk. Amazon Mechanical Turk<sup>27</sup> is an internet marketplace service for work that requires human intelligence. Five people (Wilson *et al.*, 2013) were used to classify each sentence. The people found on Amazon Mechanical Turk are

---

<sup>26</sup> [http://aclweb.org/aclwiki/index.php?title=SemEval\\_Portal](http://aclweb.org/aclwiki/index.php?title=SemEval_Portal)

<sup>27</sup> <https://www.mturk.com/mturk/welcome>

called Turkers. Turkers mark the data by using the start and end point of the phrase or word from their opinion and state whether it is negative, neutral or positive. The words that appear three times from five Turkers will be assigned labels by the organisers for each sentence.

SemEval 2013 Task 2A (Wilson *et al.*, 2013) is concerned with two subtasks: constrained (A) and unconstrained (B). A constrained task can use the data provided by the organiser while an unconstrained task can use additional data. A constrained task was chosen to avoid both resource implications and potential advantages implied by the use of additional data containing sentiment annotations. There are participants who chose the constrained task to work with additional sentiment lexicons and machine learning.

There are three sentiment lexicons and machine learning algorithms that were used the most in the constrained task of SemEval 2013 Task 2. For sentiment lexicons, they were the Bing Liu Lexicon (Hu and Liu, 2004)<sup>28</sup>, the MPQA Subjective Lexicon (Wilson *et al.*, 2005b)<sup>29</sup> and the AFINN Lexicon (Nielsen, 2011a)<sup>30</sup>. For machine learning algorithms, the most popular were the Naïve Bayes (Tan *et al.*, 2009) (Section 2.2) (Section 2.2), SVM (Kecman, 2005) (Section 2.2) and Maximum Entropy Modelling (Harte, 2011) (Section 2.2). It is not known which sentiment lexicon and machine learning algorithm achieves a better accuracy performance in the task when using the same variables and features. The details of these sentiment lexicons are in the previous section. The details of the machine learning algorithms can be found in the following section.

## 2.2 Machine Learning Algorithms

Machine learning is an area of Artificial Intelligence that is related to the study of algorithms that could be learned from data. The algorithm of machine learning is to build the model which is based on the input data and use that data to make decisions and predictions (Bishop, 2006).

Supervised learning algorithms are machine learning algorithms that classify and predict the final results by using training data that is labelled

---

<sup>28</sup> <http://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html#lexicon>

<sup>29</sup> [http://mpqa.cs.pitt.edu/lexicons/subj\\_lexicon/](http://mpqa.cs.pitt.edu/lexicons/subj_lexicon/)

<sup>30</sup> [http://www2.imm.dtu.dk/pubdb/views/publication\\_details.php?id=6010](http://www2.imm.dtu.dk/pubdb/views/publication_details.php?id=6010)

(Kotsiantis *et al.*, 2007). Labelled data can refer to data that has been classified or checked by a human and assigned a label (Zhu *et al.*, 2009a). There are three main algorithms that are commonly used in supervised learning. They are Naïve Bayes (Tan *et al.*, 2009), SVM (Kecman, 2005) and Maximum Entropy Modelling (Harte, 2011). Brief details of these algorithms are covered below.

*Naïve Bayes* (Tan *et al.*, 2009) is a classification algorithm based on Bayes' theorem that underlies the naïve assumption that attributes within the same case are independent given the class label (Elangovan *et al.*, 2010). This is also known as the 'state-of-the art' of Bayes rules (Cufoglu *et al.*, 2008). Naïve Bayes (Tan *et al.*, 2009) constructs the model by adjusting the distribution of the number for each feature. For example, in the text classification, Naïve Bayes (Tan *et al.*, 2009) regards the documents as a 'bag-of-words' and from it, the features are extracted (Liu, 2007; 2012b). Tang *et al.* (2009) considered that Naïve Bayes (Tan *et al.*, 2009) assigns a context  $X_i$  (represented by a vector  $X_i^*$ ) to the class  $C_j$  that maximises  $P(C_j|X_i^*)$  by applying Bayes's rule, as in (1):

$$P(C_j|X_i^*) = \frac{P(C_j)P(X_i^*|C_j)}{P(X_i^*)} \quad (1)$$

Source: (Tang *et al.*, 2009)

where  $P(X_i^*)$  is a randomly selected context  $X$ , the representation of vector is  $X_j^*$ .  $P(C_j)$  is the randomly selected context that is assigned to class  $C$ .

To classify the term  $P(X_i^*|C_j)$ , features in  $X_i^*$  were assumed as  $f_j$  from  $j = 1$  to  $m$  as in (2).

$$P(C_j|X_i^*) = \frac{P(C_j) \prod_{j=1}^m P(f_j|C_j)}{P(X_i^*)} \quad (2)$$

Source: (Tang *et al.*, 2009)

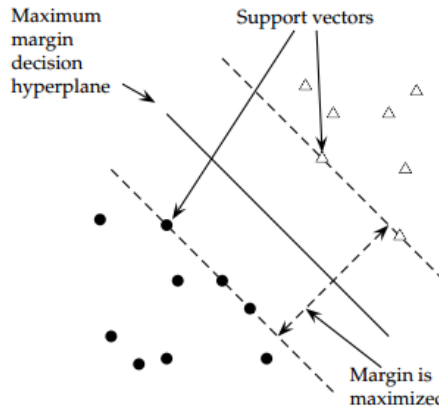
*A Support Vector Machine (SVM)* (Kecman, 2005) is a binary linear classification model with a learning algorithm for the classification and regression analysis of the data. The purpose of SVM (Kecman, 2005) is to separate datasets

into classes and to discover the decision boundary (hyperplane) for separating the dataset. In order to find the hyperplane, the maximum distance between classes (margin) will be used with the closest data points on the margin (support vector), as illustrated in Figure 3. The equation of SVM (Kecman, 2005) is presented in (3):

$$\vec{w} = \sum_j \alpha_j c_j \vec{d}_j, \quad \alpha_j \geq 0 \quad (3)$$

Source: (Kecman, 2005)

where vector  $\vec{w}$  represented as hyperplane.  $c_j$  is a polarity (negative and positive) of the data  $d_j$  which  $c_j \in \{-1, 1\}$ .  $\alpha_j$  are obtained by solving the dual optimisation problem. Vectors  $\vec{d}_j$  such that  $\alpha_j$  which are greater than zero are called support vectors, since they are the only document vectors contributing to  $\vec{w}$ . The classification of test instances consists of a simple way of determining which side of the  $\vec{w}$  hyperplane they fall on.



**Figure 2-2:** Hyperplane of support vector machine  
(Manning *et al.*, 2008b)

*Maximum Entropy Modelling (MaxEnt)* (Harte, 2011) is also known as the log-linear model and maximum likelihood exponential model (Lin *et al.*, 2008; Tsuruoka *et al.*, 2009). The MaxEnt (Harte, 2011) classification is a flexible feature-based model that aims to satisfy the constraints of available information, which also has the highest entropy. The MaxEnt (Harte, 2011) classification is implemented in a variety of ways that could be used to identify the model with the

highest entropy. The equation of MaxEnt (Nigam *et al.*, 1999; Osborne, 2002; Harte, 2011) can be presented as in (4):

$$P_{ME}(c|d) = \frac{1}{Z(d)} \exp \left( \sum_{i=1}^n \lambda_i f_i(d, c) \right) \quad (4)$$

Source: (Nigam *et al.*, 1999; Osborne, 2002; Harte, 2011)

where  $c$  is a label from the set of labels  $C$ .  $d$  is the item that we are interested in labelling from the set of training data  $S$ .  $Z(d)$  is a normalisation function. The normalisation function is normally a function that can make a wave less fuzzy, make it more like a band, and lessen the edge.  $f_i(d, c)$  is a function where some feature  $f_i$  has a weight  $\lambda_i$ .

However, this study is not concerned with the deeper questions, such as the alternatives to machine learning algorithms and the mathematics involved. Instead, the focus is on the basic knowledge required to use a machine learning algorithm to improve the performance accuracy of a supervised machine learning based classifier.

The following section provides brief details of related work that has used single machine learning algorithms for sentiment analysis.

### 2.2.1 Natural language packages with machine learning capability

The following section details the popular natural language packages that contain the abilities of machine learning algorithms.

#### I. NLTK

The Natural Language Toolkit (NLTK)<sup>31</sup> (Bird, 2006a) is widely-used machine learning open source software which was developed using Python (Python Software Foundation, 2001) and comprises the WordNet (Fellbaum, 2010; Princeton University, 2010) (Section 2.1.1.2) interface which is a lexicon database for English (Miller, 1995). NLTK (Bird, 2006a) comprises modules to access such

---

<sup>31</sup> <http://www.nltk.org/>

as corpus, tokenising, phrasing, tagging and stemming (a process to reduce the inflected word, e.g. ‘argu’ is the stemmer of ‘argue’, ‘argues’, ‘argued’ and ‘arguing’). There are machine learning algorithms that can be used in NLTK (Bird, 2006a) such as Naïve Bayes (Tan *et al.*, 2009) (Section 2.2), Maximum Entropy (Harte, 2011) (Section 2.2), Support Vector Machine (Kecman, 2005) (Section 2.2) and Decision Trees (Quinlan, 1986; Safavian and Landgrebe, 1991) (an algorithm that uses a tree graph or model to make decisions). However, the support vector machine (Kecman, 2005) has been removed from NLTK (Bird, 2006a), although it can still be used via the wrapper of Scikit-learn (Pedregosa *et al.*, 2011) in NLTK (Bird, 2006a) or directly at Scikit-learn (Derczynski, 2013).

## II. GATE

General Architecture for Text Engineering (GATE)<sup>32</sup> (Cunningham, 2002; Cunningham *et al.*, 2011) is a natural language engineering tool developed using Java, in which the GATE (Cunningham, 2002; Cunningham *et al.*, 2011) resources were released using Java beans, which is the Java framework interface. GATE (Cunningham, 2002; Cunningham *et al.*, 2011) is an open source tool that is freely available and widely used for text mining. There are three types of resource contained in GATE (Cunningham, 2002; Cunningham *et al.*, 2011). The first resource is the language resource which can refer to text documents, such as lexicons, corpora and ontologies. The text documents can be used in different formats, such as MS Word, PDF and HTML. The second resource is the processing resource which can refer to the principal programmatic resources, such as parsers, recognisers and n-gram modellers. The main processing resources of tokenizer, gazetteer, sentence splitter and part-of-speech tagger were used to create an information extraction system called ANNIE (A Nearly-New Information Extraction System) (Cunningham, 2002; Cunningham *et al.*, 2011). The final resource is a visual resource which represents visualisation and components that participate in a graphical user interface (GUI). GATE (Cunningham, 2002; Cunningham *et al.*, 2011) also has wraps for using a machine learning algorithm that is implemented in WEKA (Holmes *et al.*, 1994; Hall *et al.*,

---

<sup>32</sup> <http://gate.ac.uk/>



2009, such as Naïve Bayes (Tan *et al.*, 2009) (Section 2.2), K-Nearest Neighbour (KNN) (Altman, 1992) (non-parametric algorithm use for regression and classification) and C4.5 Decision Tree algorithm (Quinlan, 1993; Polat and Gunes, 2009) (Section 2.1.3.3).

### **III. WEKA**

Waikato Environment for Knowledge Analysis (WEKA) (Holmes *et al.*, 1994; Hall *et al.*, 2009) is a machine learning toolkit developed using Java and graphical user interfaces which are flexible and easy to use. WEKA (Holmes *et al.*, 1994; Hall *et al.*, 2009) aims to provide a collection of machine learning algorithms, such as Naïve Bayes (Tan *et al.*, 2009) (Section 2.2), Support Vector Machine (Kecman, 2005) (Section 2.2) and Logistic Regression (Kleinbaum and Klein, 2010) (Section 2.1.2), and tools for data processing by the searcher. The tools consist of the main processes for use in data mining, such as classification, clustering (used for grouping similar objects that are supposed to be in the same group (Chakraborty *et al.*, 2014)), regression (used as a measure for prediction error to model the relationship between variables that is iteratively refined), attribute selection and association rule mining. Moreover, researchers could implement the new algorithm by using the internal framework without worrying about the supporting infrastructure for data management and programme evaluation (Hall *et al.*, 2009). Witten and Frank (2011) claimed that the purpose of WEKA (Holmes *et al.*, 1994; Hall *et al.*, 2009) is to predict new instances by using a learned model (a model that has been constructed by training selected machine learning algorithms with the given datasets) to apply machine learning algorithms to the task and to compare the results. Before the input data is loaded into WEKA (Holmes *et al.*, 1994; Hall *et al.*, 2009), it must be converted to ARFF (Attribute-Relation File format) format, which is the default format, as presented in Figure 2.3.

```

@RELATION iris

@ATTRIBUTE sepallength      NUMERIC
@ATTRIBUTE sepalwidth       NUMERIC
@ATTRIBUTE class            {Iris-setosa}

@DATA
5,5,4,0,Iris-setosa
0,0,1,0, Iris-setosa
7,0,3,0, Iris-setosa

```

Figure 2-3: Example of ARFF file

#### IV. Scikit-learn

Scikit-learn (Pedregosa *et al.*, 2011) is a machine learning library that includes machine learning algorithms such as Naïve Bayes (Tan *et al.*, 2009) (Section 2.2), Support Vector Machines (Kecman, 2005) (Section 2.2), Logistic Regression (Kleinbaum and Klein, 2010) (Section 2.1.2) and Hidden Markov models (HMM) (Elliott *et al.*, 1995) (a tool for representing probability distributions over sequences of observations (Ghahramani, 2001)) as well as data processing tools, such as classification, clustering and regression tools (Pedregosa *et al.*, 2011). Scikit-learn (Pedregosa *et al.*, 2011) aims to provide better machine learning tools within programming that can be accessed by non-machine learning experts and used in scientific fields. Scikit-learn (Pedregosa *et al.*, 2011) was developed and written using Python (Python Software Foundation, 2001), although some core algorithms were written using Cython<sup>33</sup> (Behnel *et al.*, 2008; Behnel *et al.*, 2011). Cython (Behnel *et al.*, 2008; Behnel *et al.*, 2011) is a language for writing C extensions for Python (Python Software Foundation, 2001). However, Scikit-learn (Pedregosa *et al.*, 2011) depends largely on two extensions of scientific packages from Python (Python Software Foundation, 2001): Numpy (Oliphant, 2006) and Scipy (Jones *et al.*, 2001).

##### 2.2.2 Real-world Application of machine learning

There are some real-world techniques and applications that rely on machine learning algorithms; some examples of these are provided below.

<sup>33</sup> <http://cython.org/>

### **I. Search Engines**

Currently, large information collections are stored on websites, whereby the search engine could help users discover the matching information. In order to achieve this, the concept of web-page ranking has been used within the machine learning algorithms (Richardson *et al.*, 2006; Yong *et al.*, 2008). The concepts are to find the relative information in the webpages using sources, such as the contents and webpage structures. Moreover, the frequency of the suggestion links in a query that the users follow is considered. For example, Google uses the search engine based on PageRank (Page *et al.*, 1998; Langville *et al.*, 2008). PageRank (Page *et al.*, 1998; Langville *et al.*, 2008) is an algorithm used to rank the website by counting the number and quality of links to a page for determining how important the website is. On the other hand, an algorithm for calculated click-through rates for advertisement selection, called AdPredictor (Graepel *et al.*, 2010; Rowstron *et al.*, 2012) is used by Microsoft's Bing search engine.

### **II. Machine Translation**

In international companies that have multi-language partners, translated documents are important. The structure of the machine to translate is to learn a mapping of the input (A language) to output (B language); however, there are features involved (Liang *et al.*, 2006), such as spelling, part of speech and syntax (right-to-left or left-to-right language). Moreover, there are machine translators that can be used, such as Moses (Koehn *et al.*, 2007; Koehn, 2010) (statistical machine translation that allows the users to train translation models for any language pair) and IQMT (Giménez and Amigó, 2006) (a framework for automatic machine translation evaluation).

### **III. Document Categorisation**

Document Categorisation is used in fields such as computer science and information systems. Documents such as text, image and video, can be classified using features; for example, heading, subject, keywords, year, time and authors. To organise them, time and cost can be reduced by using machine learning algorithms. For example, Ballan *et al.* (2011) investigated extract actions and

events in video. Medical documents based on user profiles could be organised and extracted through the medical document index method, called AMTEX, using the MetaMap Transfer Tool (MMTX) (Hliaoutakis and Petrakis, 2011). MMTX (Hliaoutakis and Petrakis, 2011) is a tool developed for use with bibliographic material.

#### **IV. Computer vision**

Computer vision systems use machine learning algorithms to analyse, classify and understand images, such as facial and handwriting recognition. For example, in 1984, the United States Post Office used trained machine learning to sort and recognise handwritten letters (Srihari *et al.*, 1993). Moreover, Bartlett *et al.* (2005) used machine learning to detect frontal faces in video streams with respect to seven feelings and 17 action units of the Facial Action Coding System (FACS) (Ekman and Friesen, 1978). FACS (Ekman and Friesen, 1978) is an automatic system that can detect facial expressions.

#### **V. Sentiment Analysis**

The main goal of sentiment analysis is to identify the polarity of contents, in which machine learning algorithms could be used. For example, Yu and Hatzivassiloglou (2003) developed techniques based on supervised learning to classify sentence level sentiment analysis, while Esuli and Sebastiani (2005) used semi-supervised learning to determine the orientation of subjective terms. Meanwhile, Turney (2002) used unsupervised learning to classify more than 400 reviews.

There are two types of machine learning algorithms however that can be used in sentiment analysis: single machine learning algorithm and combined of machine learning algorithms (ensemble learning). More details of the single machine learning algorithms that are used in sentiment analysis are described in the following section, while the details of ensemble learning algorithms are presented in Section 2.2.3.

### 2.2.3 Sentiment analysis via single machine learning

Single machine learning algorithms have been used to analyse the sentiment of a text. For example, Pang *et al.* (2002) classified movie reviews by using only the collection of the datasets from the Internet Movie Database (IMDb) as training data, with three machine learning algorithms: Naïve Bayes (Tan *et al.*, 2009) (Section 2.2), SVM (Kecman, 2005) (Section 2.2) and Maximum Entropy Modelling (Harte, 2011) (Section 2.2). The unigram feature was tested with three machine learning algorithms for use as base results. The other features that have been used are bigrams, part-of-speech and position of text. The results showed that the use of unigrams with SVM (Kecman, 2005) (Section 2.2) achieved 82.90% accuracy while the results from Naïve Bayes (Tan *et al.*, 2009) (Section 2.2) and Maximum Entropy Modelling (Harte, 2011) (Section 2.2) achieved 81.00% and 80.40% accuracy, respectively. These base results were better than when using the combination of unigrams with bigrams (80.60%, 80.80% and 82.70% from Naïve Bayes (Tan *et al.*, 2009), Maximum Entropy Modelling (Harte, 2011) and SVM (Kecman, 2005), respectively), unigrams with part-of-speech (81.50%, 80.40% and 81.9% from three machine learning algorithms, respectively) and unigrams with the position in the text (81.0%, 80.10% and 81.6% from three machine learning algorithms, respectively). The results of this experiment showed that using a combination of features does not always achieve better accuracy than using only the unigram feature. Moreover, none of the sentiment lexicons were used in the experiment so, if the sentiment lexicons were used and merged with training datasets, would the performance achieve a better accuracy than using only the training data? This question remains unanswered.

Go *et al.* (2009) used three machine learning algorithms to classify the sentiment of tweets: Naïve Bayes (Tan *et al.*, 2009) (Section 2.2), Maximum Entropy Modelling (Harte, 2011) (Section 2.2) and SVM (Kecman, 2005) (Section 2.2). Emoticons have been used as labels (positive and negative) in training data to perform supervised learning. There are two features that were used in the experiment: unigram and part-of-speech. The results from unigram showed that, Go *et al.* (2009) achieved 81.3%, 80.5% and 82.2% from the three machine learning algorithms, respectively. In contrast, the results from the combination of

unigram and part-of-speech achieved lower accuracy at 79.9%, 79.9% and 81.9% from Naïve Bayes (Tan *et al.*, 2009), Maximum Entropy Modelling (Harte, 2011) and SVM (Kecman, 2005), respectively. Go *et al.* (2009) used a single machine learning algorithm and the combination of features but would the performance have achieved better accuracy if they had used a combination of machine learning algorithms? This question remains unanswered.

Yerva *et al.* (2010) used SVM (Kecman, 2005) (Section 2.2) to classify tweets by whether or not the context relates to the company. The dataset was obtained from WePS-3 (Natural Language Processing and Information Retrieval Group at UNED, 2010). WePS-3 (Natural Language Processing and Information Retrieval Group at UNED, 2010) is a workshop that focuses on shared tasks in the search for information about entities on the web. To solve the problem, Yerva *et al.* (2010) built a corpus by collecting keywords that were related to the company using six profiles. The first profile comprised keywords relevant to the company and were presented on the company homepage that was provided by WePS-3 (Natural Language Processing and Information Retrieval Group at UNED, 2010). The second profile, the keywords from ‘html meta tags’ (e.g. <meta>) of the webpages were collected and called the metadata profile. The third profile, Yerva *et al.* (2010) used WordNet (Fellbaum, 2010; Princeton University, 2010) (Section 2.1.1.2) to find the keywords of the category to which the company belonged and was named the category profile.

Google Sets is a source for obtaining common knowledge about a company by identifying and generating lists of the items that might be related to the company. Google Sets was used in the fourth profile for collecting the keywords related closely to the company and named as the googleset profile. The fifth and sixth profiles are collections of the keyword from users’ feedback, both positive and negative, and named as the positive profile and negative profile, respectively. After getting all the profiles, Yerva *et al.* (2010) separated the use of these profiles into four tasks: use all profiles, use all profiles except the negative feedback, use all profiles except the category profile and use only the home page. The results showed that the accuracy performance achieved an F-score of 59.50%, 62%, 60% and 48% from the four tasks respectively. In this experiment, SVM (Kecman,

2005) (Section 2.2) were used, although how much accuracy could have been achieved using the other machine learning algorithms? This question has not been answered. It should be noted that in mid-2011, Google Sets was discontinued by Google<sup>34</sup>.

Troussas *et al.* (2013) used three machine learning algorithms: Naïve Bayes (Tan *et al.*, 2009) (Section 2.2), Rocchio (Miao and Kamel, 2011) (a machine learning algorithm, but a text classifier which is based on relevance feedback) and the Perceptron (Dasgupta *et al.*, 2009) (supervised machine learning algorithms which attempt to find a hyperplane that separates two sets of points) to classify contents from Facebook using positive and negative emoticons. The datasets were collected using Facebook API<sup>35</sup>. Facebook API is a platform for building an application that is available to Facebook users. API allows the application to access to the users' information and social connections for connecting to the application for posting activities or news on users' Facebook profile pages, which is subject to the privacy settings of the users (Ortiz, 2010). The results showed that the accuracy achieved F-scores of 72%, 74% and 60% using Naïve Bayes (Tan *et al.*, 2009) (Section 2.2), Rocchio (Miao and Kamel, 2011) and the Perceptron (Dasgupta *et al.*, 2009), respectively. If the three machine learning algorithms were combined, would the accuracy performance have been better than the single machine learning algorithms? This question remains unanswered.

After reviewing the work related to machine learning algorithms in sentiment analysis, it has been found that Naïve Bayes (Tan *et al.*, 2009) (Section 2.2), Support Vector Machine (Kecman, 2005) (Section 2.2) and Maximum Entropy Modelling (Harte, 2011) (Section 2.2) are commonly used. The question arises from these machine learning algorithms: (*RQ. 5*) 'which single machine learning algorithm is essential and achieves better accuracy in the context of data classifiers?'

Therefore, for answering this question, these machine learning algorithms will be used in the TJP system (Chapter 3). They will be trained using the same variables and features and their results will be compared. Moreover, the two machine learning algorithms that achieve the most accuracy will be combined

---

<sup>34</sup> <http://googlesystem.blogspot.co.uk/2011/08/google-sets-will-be-shut-down.html>

<sup>35</sup> <https://developers.facebook.com/docs/reference/fql>

using ensemble learning with the aim of improving the accuracy. A discussion of ensemble learning can be found in the next section.

### **2.3 Ensemble Learning Algorithms; Multiple Classifiers**

Ensemble learning algorithms are an approach to machine learning algorithms where multiple classifiers are trained with the same training data and the resulting trained system is used to make the final predictions. Ensemble learning algorithms often achieve higher accuracy than using single classifiers (Rokach, 2010).

However, there is no guarantee that the ensemble learning algorithms will *always* achieve better accuracy than a single classifier (Rokach, 2010; Tang *et al.*, 2010). This given rise to: (RQ. 6) ‘If ensemble learning is used in the context of data, will the accuracy achieved be better than a single machine learning algorithm?’ Ensemble learning algorithms can be divided into two types: the common method and combining method. The details of each method are described below.

#### **2.3.1 Sentiment analysis via common methodology**

Common methods use a subset of training data for the classification system such as bagging (Breiman, 1996), boosting (Kearns, 1988) and random forest (Breiman, 2001).

*Bagging*, or bootstrap aggregation (Breiman, 1996), was the earliest and simplest ensemble algorithm (Breiman, 1996). This method of classifier is built by using a random subset of training data. The output of the models is taken as the majority vote from each classifier (Sewell, 2008). For example, Qadir and Riloff (2013) used a bagging algorithm (Sun and Pfahringer, 2011) to classify five classes of hashtags that have been used on Twitter. The classes are: affection, anger/range, fear/anxiety, joy and sadness/disappointment. Qadir and Riloff (2013) manually selected hashtags that are defined as representative of the emotion for each class. The bagging algorithm (Sun and Pfahringer, 2011) was used to learn 10 hashtags for hundred iterations; this data will be used to perform a ‘list lookup’ for searching the seed hashtag within tweets and assigning the label. The gold standard for the dataset was annotated by two annotators.



Krippendorff's alpha (Krippendorff, 2011) was used to measure the agreement. This produced a Kappa value of 0.79 which is an acceptable agreement. The results from using the combination of a unigram and list lookup achieved an F-score of 61%, 44%, 54%, 59% and 46% for each group, respectively. On the other hand, the results from using the combination of unigram and seed list-lookup achieved an F-score of 54%, 30%, 44%, 56% and 40% of each group, respectively.

*Boosting* (Kearns, 1988) is a process where the training subset of each classifier is chosen based on the performance of the classifier that has previously been trained (Schapire, 1990). The model that has been misclassified will give higher weight than the correct one. The majority votes from each classifier also be used for creating the output as in Bagging (Sewell, 2008). For example, Celikyilmaz *et al.* (2010) classified the sentiments of tweets based on two groups: polar and non-polar. The tweets that have positive or negative sentiments were labelled 'polar'; otherwise, a 'nonpolar' label was used. To reduce the sparseness caused by noise in the tweets, the pronunciation of words was used to map alternative and shorter spelling into the intended words. The tweet collections from September 2009 to June 2010 were selected by using a keyword search of products and organisations to identify tweets that were related to the mobile operation.

Latent Dirichlet Allocation (LDA) (Blei *et al.*, 2003) was used to capture the polarity of words generating the semantic content of tweets. Celikyilmaz *et al.* (2010) use LDA (Blei *et al.*, 2003) to extract semantic concepts in tweets as probability distributions over words that tend to co-occur within text, without using any non-polar tweets. LDA (Blei *et al.*, 2003) is a topic model based on the probabilistic approach. The data was then separated into two sets: training and testing data. Celikyilmaz *et al.* (2010) wrote that Set-1 was a set in which the training data was used with n-fold cross validation (Section 2.1.1.3). On the other hand, Set-2 was a set that used both training and testing data. There were two main approaches in the experiment. The first approach was to test the effect of using part-of-speech classes for text normalisation of tweets. Celikyilmaz *et al.*

(2010) detected polarity with all word unigrams, bigrams and trigrams as features by using BoosTexter (Schapire and Singer, 2000). BoosTexter (Schapire and Singer, 2000) is a general purpose machine learning program based on a boosting (Kearns, 1988) algorithm which can handle multi-class problems such as unbalances in different classes of the data. The results revealed that, using clustering to determine the true grouping of the words with the same pronunciation helps to reduce error rate and improve the F-measure from 43.1% to 47.7% and 35.4% to 35.6% on Set-1 and Set-2, respectively. The final approach was to use lexicons that were extracted using LDA (Blei *et al.*, 2003) to classify the polarity of tweets. Similar to the previous approach, two versions of BoosTexter (Schapire and Singer, 2000) were used. The results showed that the performance of Set-1 retains a higher accuracy than Set 2 at an F-measure of 47.4%.

Bermingham and Smeaton (2011) monitored political sentiment and predicted election results using Twitter. The datasets was collected from tweets which were written when the Irish General Election took place in 2011. The tweets that were related to five parties were selected: Fianna Fail (FF), the Green Party, Labour, Fine Gael (FG) and Sinn Fein (SF). The tweets were identified using keywords of parties' names, abbreviations and hashtags (#ge11). However, the independent candidates or the minority parties were not included. MAE (Willmott and Matsuura, 2005) (Section 2.1.2) was used to compare tweets' base predictions with polls as indicators for the election's results. Nine polls that were commissioned during the election were used to provide the reference point of the analysis. Bermingham and Smeaton (2011) stated that a 3% accuracy of the polls is guaranteed by their method and the average of MAE (Willmott and Matsuura, 2005) (Section 2.1.2) appeared as 1.61% after being compared with the final election results based on the five parties.

To analyse the sentiments of the tweets, nine annotators were used. Bermingham and Smeaton (2011) wrote that annotators were instructed not to consider reporting of positive or negative facts as sentiments but that sentiment needed to be one of emotion, opinion, evaluation or speculation towards a subject. However, tweets were classified into three polarities: positive, negative and

mixed. Their agreement was measured using Krippendorff's Alpha (Krippendorff, 2011) (Section 2.1.1.4). Krippendorff's Alpha was 0.478. Bermingham and Smeaton (2011) used the boosting (Kearns, 1988) approach from Adaboost (Freund and Schapire, 1996) to classify the sentiments of tweets with 10 training iterations as implemented in WEKA (see Section 2.2.1). Two versions of Adaboost (Freund and Schapire, 1996) were used and the results revealed that the Adaboost MNB classifier (Section 2.1.3.3) achieved better accuracy than the Adaboost SVM classifier (Section 2.2) with average F-scores of 65.09% and 64.28% respectively within the 10-fold cross-validation (Section 2.1.1.3) for the three polarities.

*Random Forest method (Breiman, 2001)* is built based on decision trees (Section 2.1.3.3) and is the implementation of the random subspace method (Breiman, 2001). The random subspace method used the data in the same way as in bagging but uses the feature instead of the data (Ho, 1998). For example, Liu et al. (2013) determined sentiments from tweet events. The dataset was from Liu et al. (2013)'s database and Text REtrieval Conference (TREC) 2011: Microblog Track, which focused on four events: US unemployment, American railway services, the BBC World Service staff cuts and the election of President Obama between June 2008 and May 2009. There are two groups of features that were used. The first was the textual feature. The textual feature used a word-based review in which texts were evaluated by the sentiment orientation of extracted phrases using PMI-IR (Turney, 2001) (Section 2.1.1.1). Next, WordNet Affect (Strapparava and Valitutti, 2004) (Section 2.1.1.4) was used to assign the polarity of texts. The final feature was non-textual. A non-textual feature refers to emoticons, temporal features and punctuation. An emotions dictionary was built using a collection of emoticons from Wikipedia. Temporal features referred to the time that the tweets were posted: hours, dates, the day of week and month by Liu et al. (2013). Punctuation marks such as exclamation (!) and question marks (?) were used in the non-textual feature.

In order to classify the sentiment polarity, SVM (Kecman, 2005) (Section 2.2), Random Forest (Breiman, 2001) classifiers and co-training methods were

used. Co-training is a labour-intensive task which manually labels a large number of tweets (Liu *et al.*, 2013). The accuracy performance showed that the results from the co-training methods achieved higher F-scores than others at 80.2%, 81.6%, 83.2% and 81.1%, in all five events, respectively. The results from the random forest (Breiman, 2001) method were better than SVM (Kecman, 2005) (Section 2.2) with an F-score of 78.9% and 80.6% in the events of US unemployment and the BBC World Service staff cuts, respectively. In contrast, the results from SVM (Kecman, 2005) (Section 2.2) were better than the random forest (Breiman, 2001) with an F-score of 80.4% in the event of American railway services. In addition to these results, the results from the random forest (Breiman, 2001) and SVM (Kecman, 2005) were equal with an F-score of 78.10% in the event of the Obama Election. Furthermore, Liu *et al.* (2013) determined the sentiments using graph methods that were used to display the time series of sentiment word labels, named visualisation graphs. However, Liu *et al.* (2013) only discussed the results from the Obama Election. The graphs shown demonstrated that people's sentiments about Obama fluctuated over time, especially when influential events occurred. Furthermore, this example demonstrated that an ensemble learning algorithm does not always achieve better accuracy performance than a single machine learning algorithm.

Siswanto and Khodra (2013) developed a system to predict the latent attributes of tweets. A Latent attribute refers to an attribute that has not been stated clearly or directly, for example, gender, age and origin (Rao *et al.*, 2010). Siswanto and Khodra (2013) however only focused on age (under or over 20 years old) and occupation (student or employee). The dataset was only collected from Twitter in the Indonesian language. The common words of each category (under or over 20 years old and student or employee) were focused on and used for building the corpus. For example, a user who is a student (category) uses words that are related to school or college. In contrast, employees (category) use words that related to jobs or processes. Once the dataset and lexicon were established, they were passed to pre-processing process using eight features: retweets, mention/links, duplicate letters, numbers, stopwords, punctuation removal, converting emoticons by adopting the labels from (Sunni and

Widyantoro, 2012) and converting to lowercase. Next, the datasets were converted to ARFF (Section 2.2.1) before training in WEKA (Hall *et al.*, 2009) (Section 2.2.1). Three machine learning algorithms were used: Naïve Bayes (Tan *et al.*, 2009) (Section 2.2), SVM (Kecman, 2005) (Section 2.2) and Random Forest (Breiman, 2001). The results of each classifier were compared based on both attributes: age and job. For age classification, the results of SVM (Kecman, 2005) (Section 2.2) yielded better accuracy than the others at 77.27%. By contrast, the results from Random Forest (Breiman, 2001) (Section 2.2) achieved the best accuracy at 73.08% for the classification of jobs. The results from Naïve Bayes (Tan *et al.*, 2009) (Section 2.2) achieved a lower accuracy in both attributes at 66.36% and 70.19% in age attribute and job attribute, respectively.

In addition to the common methods, there is another method in the ensemble learning algorithm, the combining method. Combining methods are the methods that use the combination of two or more machine learning algorithms for classification. The methods of combining approaches can be divided into two types: simple combining and meta-combining methods. The details of the simple combining method are described in the following section, while the details of meta-combining methods are detailed in Section 2.3.3.

### **2.3.2 Sentiment analysis via simple combining methodology**

Simple combining methods combine the outputs from multiple classifiers to provide the classification; for example, majority voting and weighted voting.

*Majority voting* (Polikar, 2012) is a basic and simple algorithm that uses a combination of classifiers. The decisions of the voting depend on the agreement between more than half of the classifiers; otherwise, the input is rejected.

For example, Gryc and Moilanen (2014) analysed sentiment from a dataset around the 2008 U.S. Presidential Election, collected by IBM's Predictive Modelling Group. The data was labelled as positive, neutral, negative and not applicable with the respect to Barack Obama using Amazon's Mechanical Turk (Section 2.1.4). However, only positive, neutral and negative labels were focused on by Gryc and Moilanen (2014). Moreover, three features were used. The first feature was the unigram bag-of-words. Second, the social network feature which

use the context from social networks and blogs as the classification for producing the features. The final feature was the sentiment analysis feature which referred to the sentiment scoring for the classification. Two single machine learning algorithms were used: Naïve Bayes Multinomial (NBM) (McCallum and Nigam, 1998a; Bermejo *et al.*, 2011) (Section 2.1.3.3) and Logistic Regression (Kleinbaum and Klein, 2010) (Section 2.1.2) which is a method of statistical classification that is used to predict the data from the binary predictor. In addition, two ensemble learning algorithms were also used: Stacking (Wolpert, 1992) (Section 2.4.3) and Majority Voting (Polikar, 2012).

There were six performance tests of those three different features. The first performance, the social network feature, was used with Logistic Regression (Kleinbaum and Klein, 2010) (Section 2.1.2), and named as SNA. The second performance, the sentiment analysis feature was used with NBM (McCallum and Nigam, 1998a; Bermejo *et al.*, 2011) (Section 2.1.3.3), and named as, SA. The third performance, the unigram bag-of-words feature was used with NBM (McCallum and Nigam, 1998a; Bermejo *et al.*, 2011) (Section 2.1.3.3), and named as BOW. For the forth performance, all features were used with NBM (McCallum and Nigam, 1998a; Bermejo *et al.*, 2011) (Section 2.1.3.3), named as, ALL. The three separate NBM classifiers NBM (McCallum and Nigam, 1998a; Bermejo *et al.*, 2011) (Section 2.1.3.3) from SNA, SA and BOW were used with Stacking (Wolpert, 1992) (Section 2.4.3) and Majority Voting (Polikar, 2012), named as STACK and VOTE respectively, as the last two performance tests. The results achieved 36.30%, 44.63%, 48.41%, 47.72%, 44.33% and 46.68% for SNA, SA, BOW, ALL, STACK and VOTE respectively. The results showed that STACK achieved lower accuracy than when using a single machine learning algorithm and vice versa for VOTE.

It can be said that there is a chance that the use the use of ensemble learning does not always improve the accuracy performance. Gryc and Moilanen (2014) used Stacking (Wolpert, 1992) and Majority Voting (Polikar, 2012) with three separate groups of NBM classifiers (McCallum and Nigam, 1998a; Bermejo *et al.*, 2011) (Section 2.1.3.3). but how much accuracy performance will achieve if stacking (Wolpert, 1992) and majority voting (Polikar, 2012) were used with ALL

and three separate group of NBM classifier (McCallum and Nigam, 1998a; Bermejo *et al.*, 2011) (Section 2.1.3.3)? Moreover, will a better accuracy be achieved than the results from the single machine learning algorithms? These questions have still not been answered.

*Weighted voting* is a method where each classifier is assigned a weight based on the performance of each classifier. The member's weight indicated each classifier's effect on the final classification. The assigned weight could be fixed or dynamically determined for the specific instance to be classified (Rokach, 2009). For example, a Weighted Voting approach was used to extract events and identify the relationship between the event's time and event's document creation time by Kolyal *et al.* (2013). The datasets were taken from TempEval-2<sup>36</sup> and based on the TimeBank corpus. The TimeBank corpus is a gold standard that was annotated using the TimeML mark-up scheme. TimeML is a general multilingual mark-up language for temporal information in texts.

Within the three main tasks: event extraction, event's document creation time (DCT) relation identification and event time relation identification, different features were assigned before the training with machine learning algorithms. For the event extraction, seven syntactic features were used including: part-of-speech of the event, tense (e.g. present, past), aspect of the event, polarity of the event (which will assigned as negative if the event instance is negated and vice versa for the positive polarity), modality feature (which is used only if there is a modal word that modifies the instance), event class (e.g. action of person or an organisation) and event stream that is used to stream the main event. Then, non-verbal event nouns such as war, attempts and tours were identified using WordNet (Fellbaum, 2010; Princeton University, 2010) (Section 2.1.1.2). Moreover, the Stanford Named Entity (NE) tagger was used for tagging a person, location, organisation and others. Finally, a semantic role label (Gildea and Jurafsky, 2002) was used to identify different features of the sentences of a document to help extract the events. For the event document creation time (DCT) relation identification, two features were used. Firstly, four syntactic features were used

---

<sup>36</sup> <http://timeml.org/tempeval2/>

composed of part-of-speech, tense, aspect and temporal relation between the DCT and the temporal expression in the target sentence (e.g. greater than, less than, equal to or noun). Secondly, the derived feature was used to identify the different types of context-based syntactic features derived from the text to distinguish the different types of temporal relations. For the event time relation identification, the same syntactic features and derived features as in event DCT relation extraction, event time and strings were used. Moreover, the different types of context-based temporal expression features were also identified.

Once these features were used with the dataset in three tasks, they were trained to machine learning algorithms: SVM (Kecman, 2005) (Section 2.2) and Conditional Random Field (CRF) (Lafferty *et al.*, 2001) which is an undirected graphical model that corresponds to conditionally-trained probabilistic finite state automata (Kolyal *et al.*, 2013). Next, their results were combined using Majority Voting (Polikar, 2012) and Weighted Voting for determining the final classifier. For using these voting approaches, Kolyal *et al.* (2013) defined the Majority Voting by assigning the same voting weight in the model and proposed the majority model by combining the systems. On the other hand, for Weighted Voting, the F-measure of each classifier was used as the weight of the corresponding classifier. The results of event extraction achieved F-measures of 83.54%, 83.94%, 84.65% and 85.50% for CRF (Lafferty *et al.*, 2001), SVM (Kecman, 2005) (Section 2.2), Majority Voting and Weighted Voting, respectively. The results of the event DCT relation extraction achieved F-measures of 83.60%, 82.90%, 84.10% and 84.90% for CRF (Lafferty *et al.*, 2001), SVM (Kecman, 2005) (Section 2.2), Majority Voting and Weighted Voting, respectively. The results of the event time relation extraction achieved F-measures of 64.90%, 63.80%, 65.40% and 65.90% for CRF (Lafferty *et al.*, 2001), SVM (Kecman, 2005) (Section 2.2), Majority Voting and Weighted Voting, respectively. Overall, from the three tasks, the performance of Weighted Voting achieved the best accuracy. However, Kolyal *et al.* (2013) did not explain clearly how they assigned and used Majority Voting in the experiment.

The following section provides a brief outline of meta-combining methods.



### 2.3.3 Sentiment analysis via meta-combining methodology

Meta-combining methods refer to the classifiers that are produced by inducers and from the classifications of thesis classifiers in training data. Methods used in meta-combining include Stacking, Grading, Arbiter Tree and Combiner Tree.

*Stacking* (Wolpert, 1992) is a technique that uses two classifier levels: a base classifier and meta-classifier. The process of stacking (Wolpert, 1992) is that, the output from classifier 0 will be used to train classifier 1 to predict the final output. For the formal description of stacking, Wolpert (1992) stated that the idea of stacking is to use the output from the base classifier as input for the meta-classifier to produce the final prediction. The idea of stacking (Wolpert, 1992) is that, when given a dataset  $L = \{(y_n, x_n), n = 1, \dots, N\}$ , where  $y_n$  is the class value and  $x_n$  is a vector representing the attribute values of  $n$  instance, randomly split the data into  $J$  sets. Define  $L_j$  and  $L^{(-j)} = L - L_j$  to be the testing and training set for  $j$  folds of  $J$ -fold cross validation. Given  $K$  learning algorithms, which are called, *level – 0 generalisers*, produce a  $k$  algorithm for the data in training set  $L^{(-j)}$  to produce a model  $M_k^{(-j)}$ , for  $k = 1, \dots, K$  which are called, *level – 0 models*.

For each instance  $x_n$  in  $L_j$ , the test set for  $J$ -fold cross validation, let  $z_{kn}$  mean the prediction of model  $M_k^{(-j)}$  on  $x_n$ . At the end of the cross-validation process, the dataset assembled from the output of  $K$  models is  $L_{cv} = \{(y_n, z_{1n}, \dots, z_{Kn}), n = 1, \dots, N\}$ . This is the *level – 1 data*.

The *level – 1 generaliser* is a learning algorithm derived from the data of model  $\widetilde{M}$  for  $y$  as a function of  $(z_1, \dots, z_K)$ . These are called a *level – 1 model*. To complete the training process, the final level-0 models  $M_k, k = 1, \dots, K$  are derived using all the data in  $L$ . To consider the classification process, which used the model  $M_k, k = 1, \dots, K$ , in conjunction with  $\widetilde{M}$ . Given a new instance, model  $M_k$  produces a vector  $(z_1, \dots, z_k)$ . This vector is input to the level-1 model  $\widetilde{M}$ , whose the output is the final prediction of the results of the instance.

For example, Martin-Valdivia *et al.* (2013) classified film reviews that were collected from the MuchoCine website using two combined techniques: Majority

Voting (Polikar, 2012) (Section 2.3.2) and Stacking (Wolpert, 1992). There were two parts to the experiment. For the first part, SentiWordNet (Baccianella *et al.*, 2010a) (Section 2.1.3.6) and SVM (Kecman, 2005) (Section 2.2) were used as base classifiers. The result were used in Majority Voting (Polikar, 2012) to perform the final prediction. For the second part, Stacking (Wolpert, 1992) was used with four machine learning algorithms: SVM (Kecman, 2005), Naïve Bayes (Tan *et al.*, 2009) (Section 2.2), C4.5 Decision Tree algorithm (Quinlan, 1993; Polat and Gunes, 2009) (Section 2.1.3.3) and Bayesian Logistic Regression (BLR) (Hosmer Jr *et al.*, 2013). BLR (Hosmer Jr *et al.*, 2013) is different from standard Logistic Regression (Kleinbaum and Klein, 2010) (Section 2.1.2) by assuming the model parameters are random variables. The results of Stacking (Wolpert, 1992) achieved an F-score of 88.56%, which was slightly higher than Majority Voting (Polikar, 2012) at an F-score of 88.28%.

*Grading* (Seewald and Fürnkranz, 2001; Witten and Frank, 2005) is a method that uses grades to label the predictions from the base classifier as correct (graded +) or incorrect (graded -). The prediction that contains the highest score is chosen as the final decision (Seewald and Fürnkranz, 2001). The process of Grading (Seewald and Fürnkranz, 2001; Witten and Frank, 2005) is that for each base classifier, one meta-classifier is learned whose task is to classify when the base classifier misclassifies. At the time of the classification, each base classifier classifies the unlabelled instance. The final classification is derived from the classifications of those base classifiers that are classified to be correct according to the meta-classification schemes (Rokach, 2005). The approach of Grading (Seewald and Fürnkranz, 2001; Witten and Frank, 2005) is that, level-1 classifiers exist to correct potential false decisions of level-0 classifiers (Lingenfelser *et al.*, 2011). Furthermore, Grading (Seewald and Fürnkranz, 2001; Witten and Frank, 2005) is different from Stacking (Wolpert, 1992), where the former classifier does not change the instance attributed by replacing them with class predictions or class probabilities; rather, it modifies the class values (Lingenfelser *et al.*, 2011).

For example, Lingenfelser *et al.* (2011) detected sentiment within visualisation by using the process that was adopted from meta-classification:

Stacking (Wolpert, 1992) and Grading (Seewald and Fürnkranz, 2001; Witten and Frank, 2005). The dataset was used from two corpora: DaFEx (Battocchi *et al.*, 2005) and CALLAS (Caridakis *et al.*, 2010). Both contained audio-visual recordings of Italians speaking. Lingenfelser *et al.* (2011) extracted acoustic features related to the paralinguistic message of speech from the audio channel. Videos were analysed using SHORE (Küblbeck and Ernst, 2006), a library for facial emotion detection. For recording, analysing and recognising human behaviour in real-time, a Social Signal Interpretation (SSI) (Wagner *et al.*, 2011) framework was used. However, the classification tasks were done using a Naïve Bayes (Tan *et al.*, 2009) (Section 2.2) before being combined with the meta-classifier. The results of DaFEx (Battocchi *et al.*, 2005) obtained an average of 52% and 54% using Stacking (Wolpert, 1992) and Grading (Seewald and Fürnkranz, 2001; Witten and Frank, 2005), respectively. On the other hand, The results of CALLAS (Caridakis *et al.*, 2010) obtained an average of 60% and 55% by using Stacking (Wolpert, 1992) and Grading (Seewald and Fürnkranz, 2001; Witten and Frank, 2005), respectively. From these results it can be said that, there is no guarantee that the same algorithms will achieve the same performance when the environment is changed and in relation to this experiment, the environments are refer to the two corpora.

Besides Grading (Seewald and Fürnkranz, 2001; Witten and Frank, 2005), there are other meta-combining methodologies. These are the Arbiter Tree (Chan and Stolfo, 1993) and Combiner Tree (Chan and Stolfo, 1997) methods. The Arbiter Tree (Chan and Stolfo, 1993) is a method that uses training data output that has been classified using base classifiers with selection rules. Selection rules are used for comparing the prediction of base classifiers for choosing the training dataset for the arbiter. Next, the final prediction is decided upon according to the base classifiers and arbiter by using arbitration rules with the aim of learning from incorrect classifications (Chan and Stolfo, 1993). Meanwhile, the Combiner Tree (Chan and Stolfo, 1997) is a method similar to Arbiter Tree; however, the Combiner Tree (Chan and Stolfo, 1997) will be trained directly by the training output from the base classifiers which passed the composition rules. Subsequently, the final prediction will be classified by the combiner. There are

two versions of composition rules. The first version uses the combination of results from the base classifier, while the second version also uses training data. The aim of the Combiner Tree (Chan and Stolfo, 1997) is to learn from correct classification (Chan and Stolfo, 1997) (For more details of these see Chapter 4). Related work that has used the Arbiter Tree (Chan and Stolfo, 1993) and Combiner Tree (Chan and Stolfo, 1997) algorithms in the area of sentiment analysis has not been found. Consequently, it may be assumed that neither of them have not been used previously. Therefore, the following questions arise: (*RQ. 7*) ‘Will the Arbiter Tree (Chan and Stolfo, 1993) and Combiner Tree (Chan and Stolfo, 1997) achieve better performance in the sentiment task than a single machine algorithm? And, (*RQ. 8*) ‘When comparing the Arbiter Tree (Chan and Stolfo, 1993) and Combiner Tree (Chan and Stolfo, 1997) with the other algorithms in ensemble learning, which will achieve the better performance?’

## **2.4 Conclusion**

In this chapter, the basic terms of sentiment analysis have been described. Following the level, purpose and goal of sentiment analysis is to determine the polarity of words, phrases, sentences and documents. An overview of comparative assessment in which we participated is presented in Section 2.1.4. In Section 2.2, outlines of single machine learning algorithms have been mentioned with related works. Besides that, there are brief details of ensemble learning algorithms in Section 2.3. The methods of ensemble learning can be divided into two types: common and combined methodologies. Common methods are the methods that use the subset of training data or features for the classification system. On the other hand, combined methods are the methods that used the combination of two or more than two machine learning algorithms for classification. There are two types of combined methods: simple combining and meta-combining methodologies. Simple combined methods use the combination of the output from multiple classifiers to provide the classification while meta-combining methods use a classifier to learn and decide the final classification from the output of the single classifier. The following chapters will attempt to answer the questions outlined below and describe.

- RQ 1.** How much accuracy in the context of data will be achieved when using SentiStrength (Thelwall *et al.*, 2010b) and SentiWordNet (Baccianella *et al.*, 2010a)? Moreover, will the accuracy be better than the results from word polarity (positive and negative)? (as indicated in Section 2.1.3.6)
- RQ 2.** Are sentiment lexicons essential in sentiment analysis? (as indicated in Section 2.1.3.7)
- RQ 3.** How much accuracy will be achieved in the contexts of data if using only training data? (as indicated in Section 2.1.3.7)
- RQ 4.** Will the accuracy improve if using the combination of training data and sentiment lexicon(s)? (as indicated in Section 2.1.3.7)
- RQ 5.** Which single machine learning algorithm is essential in the context of data classifiers between Naïve Bayes (Tan *et al.*, 2009), Support Vector Machine (Kecman, 2005) and Maximum Entropy Modelling (Harte, 2011)? (as indicated in Section 2.2.3)
- RQ 6.** If the ensemble learning is used in the context of data, will the accuracy achieved be better than single machine learning? (as indicated in Section 2.3)
- RQ 7.** Will Arbiter Tree (Chan and Stolfo, 1993) and Combiner Tree (Chan and Stolfo, 1997) achieve better performance in the sentiment task than the single machine algorithm? (as indicated in Section 2.3.3)
- RQ 8.** ‘When comparing the Arbiter Tree (Chan and Stolfo, 1993) and Combiner Tree (Chan and Stolfo, 1997) with the other algorithms in ensemble learning, which will achieve the better performance?’ (as indicated in Section 2.3.3)

The next chapter describes and designed the system built for the experiments in Chapter 4 and 5

## Chapter 3 : System Design

This chapter describes the TJP<sup>37</sup> system that is allowed to systematically vary factors such as, features, dataset, sentiment lexicons, sentiment resources and machine learning algorithms, to identify the interaction between and to compare components. The structure of this chapter is as follows: Section 3.1 presents and explains the user interface of TJP. Next, the system architecture and components are presented and described in Section 3.2. Finally, the series of actions for using the TJP are shown in Section 3.3.

### 3.1 System User Interface

The user interface is illustrated in Figure 3.1. The interface is simple because the evaluations of the system's outputs are focused rather than creating a user-focused interface for a finished system. The top box on the left of the user interface shows the data and lexicons that are used as training data. There are six square buttons; more than one item of training data can be chosen and they can be used in combination. For more details of the lexicons, see Sections 3.2.2 and 3.2.3. On the other side of the user interface is the testing data (Section 3.2.1), which can be used one at a time.

Next, the users can choose to train the data directly to the machine or pass the data to pre-processing (Section 3.2.4) by choosing the features. Multiple features can be chosen. After that, if the user chooses to train the data with sentiment resources, all of the training data selections will be removed automatically. The reason for this is that only testing data will be trained directly with sentiment resources, as described in Section 3.2.5. On the other hand, both training and testing data are used with supervised learning algorithms. They are the machine learning algorithm and ensemble learning algorithm (Section 3.2.6). Finally, there is a 'Submit' button and, after that is pressed, the output will be produced and generated using an evaluation method (Section 3.2.7). The details of the TJP system are explained in depth in the following section.

---

<sup>37</sup> These are the initials from the first names of the author and the supervision team

TJP's Sentiment System	
<b>Training data</b> <input checked="" type="checkbox"/> Twitter training data (TR) <input type="checkbox"/> Hu and Liu's lexicon (HL) <input type="checkbox"/> MPQA's lexicon (MPQA) <input type="checkbox"/> AFINN's lexicon (AFINN) <input type="checkbox"/> SentiWordNet's lexicon (SWN) <input type="checkbox"/> SentiStrength's lexicon (SS)	<b>Testing data</b> <input type="radio"/> Twitter <input type="radio"/> SMS
<b>Pre-processing</b> <div> <input checked="" type="radio"/> Raw (no features)  <input type="radio"/> Use Feature           <div> <input type="checkbox"/> Label emoticons  <input type="checkbox"/> Convert negation  <input type="checkbox"/> Remove @user and URLs  <input type="checkbox"/> Convert #hashtag             </div> </div> <div> <input type="checkbox"/> Reduce repeated letter  <input type="checkbox"/> Convert slang  <input type="checkbox"/> Remove stopwords  <input type="checkbox"/> Remove special characters           </div>	
<b>Sentiment Resources</b> <input type="radio"/> SentiWordNet <input type="radio"/> SentiStrength	
<b>Supervised Learning</b> <div> <b>Ensemble Learning Algorithm</b>  <input type="radio"/> Majority Voting  <input type="radio"/> Stacking  <input checked="" type="radio"/> Arbiter Tree  <input type="radio"/> Combiner Tree           </div> <div> <b>Machine Learning Algorithm</b>  <input type="radio"/> Naive Bayes  <input type="radio"/> Support Vector Machine  <input type="radio"/> Maximum Entropy Modelling           </div>	
<input type="button" value="Submit"/>	

Figure 3-1: User interface of TJP system

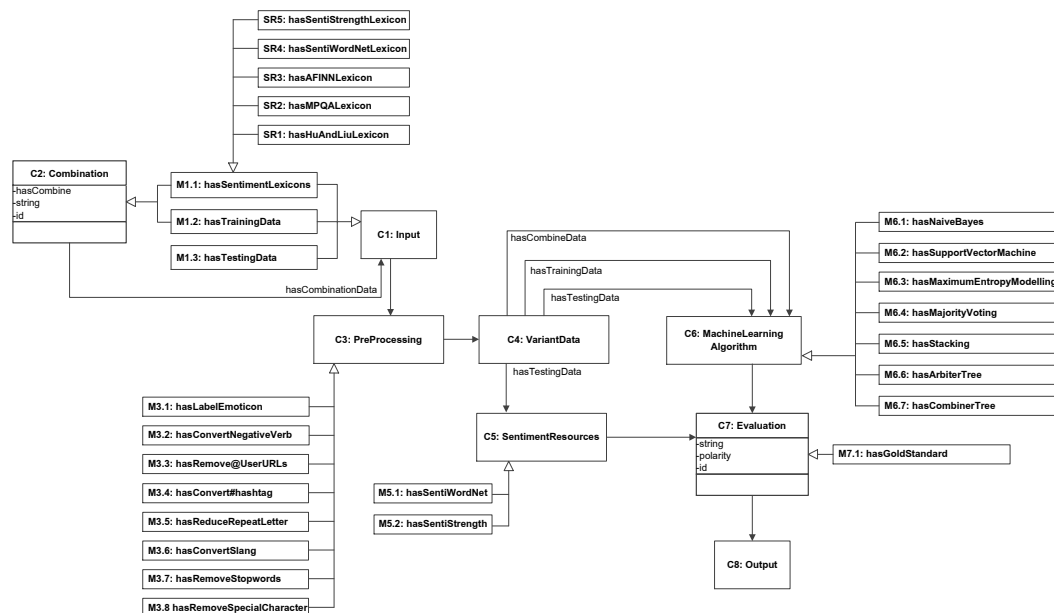


Figure 3-2: Simple UML class diagram of TJP system

### 3.2 System Architecture

Figure 3.2 shows the relationships between the different classes and models that are required for building the TJP system. ‘C1: Input’ class provides a low-level interface for inputting data into the system. The data inputs are separated into three modules: ‘M1.1: hasTrainingData’ (Section 3.2.1), ‘M1.2: hasTestingData’ (Section 3.2.1) and ‘M1.3 hasSentimentLexicons’ (Section 3.2.2). Moreover, ‘C2: Combination’ class is used for combining ‘M1.1: hasTrainingData’ with ‘M1.3 hasSentimentLexicons’, if these are needed before inserting the data input, as described in Section 3.2.3. After that, the data passes to ‘C3: pre-processing’ which controls all the features, as described in Section 3.2.4. ‘C4: VariantData’ class contains the variants of the data input from ‘C3: PreProcessing’. ‘C5: SentimentResources’ is composed of ‘M5.1: hasSentiWordNet’ and ‘M5.2: hasSentiStrength’ which used only testing data in the classification, as described in Section 3.2.5. ‘C6: MachineLearningAlgorithm’ consists of ‘M6.1: hasNaiveBayes’, ‘M6.2: hasSupportVectorMachine’, ‘M6.3: hasMaximumEntropyModelling’, ‘M6.4: hasMajorityVoting’, ‘M6.5: hasStacking’, ‘M6.6: hasArbiterTree’ and ‘M6.7: hasCombinerTree’. The details of ‘M6.1 to M6.3’ are described in Section 3.2.6 while the details of ‘M6.4 to M6.7’ are mainly described in Chapter 5. ‘C7: Evaluation’ class is used for evaluating the results from ‘C5: SentimentResources’, ‘C6: MachineLearningAlgorithm’ and ‘M7.1: hasGoldStandard’ for generating the final output in ‘C8: Output’, as described in Section 3.2.7 and Section 3.2.8, respectively.

#### 3.2.1 Data Input

All data input is handled by ‘C1: Input’ class. Data input files are in plain text format encoded in UTF-8. Only the datasets (Tweets and SMS) that were received from SemEval 2013 Task 2A (Wilson *et al.*, 2013) were used in this thesis. No additional data was collected from Twitter or elsewhere. This has the advantage of being both publicly available, and used by several other researchers to allow for the comparison of results. SemEval 2013 Task 2A (Wilson *et al.*, 2013) includes two datasets; Tweets and SMS. The dataset is made up of Tweets and SMS. The



Tweets were collected from Twitter over one-year period spanning from January 2012 to January 2013 by using the Twitter API (Wilson *et al.*, 2013).

There are some concerns regarding the ethical use of Tweets in research. The statements in Twitter's current Privacy Policy (that was amended 24<sup>th</sup> January 2016, Appendix VI) imply that, if users have shared something publicly via Twitter, their Tweets can be used for research and by third-parties. Hence, the users who tweeted before this policy was changed and who do not want their Tweets to be used by third-parties or in research, could delete their Tweets if they do not agree with this recently changed policy.

The published ethical guidelines from Rivers and Lewis (2014) suggested that it is ethical to collect information from Twitter but it is unethical to release the identification information of the authors without consent. Therefore, it is important for the researchers to protect and consider users' privacy when using Tweets in their research (Moyer, 2014; Rivers and Lewis, 2014; Trent, 2014).

This is the reason that SemEval 2013 Task 2A (Wilson *et al.*, 2013) provided a python script to download data instead of giving us the original Tweets. Therefore, our work is retrospectively conformant with Rivers and Lewis (2014) ethical guidelines since the tweets are anonymous and screen names are not used.

Consequently our work follows Twitter's Privacy Policy (Twitter, 2016) and is ethical, since all Tweets are anonymised and only used in statistical analysis. Thus the individual use and copyright are both respected. The only username used in this thesis is that of the author (@tawunrat) in Section 3.2.4.

For SMS data, SemEval 2013 Task 2A (Wilson *et al.*, 2013) used the data from the NUS SMS corpus (Chen and Kan, 2013). The organiser of SemEval 2013 Task 2A (Wilson *et al.*, 2013) annotated Both Tweets and SMS data using the Amazon Mechanical Turk (Section 2.1.4). For each sentence, five Amazon Turk workers ('Turkers', Section 2.1.4) marked the start and end point in their opinion for the phrase or word, and stated whether it was negative, neutral or positive. The words that appeared three times from five Turkers were assigned labels by the organisers for each sentence. The purpose of having a separate test

set of SMS messages is to observe how generalisable the systems trained on Twitter data are for other types of message data.

The Tweet data from SemEval 2013 Task 2A (Wilson *et al.*, 2013) contains training data (8,243 contexts<sup>38</sup>) (*'M1.1: hasTrainingData'* module), testing data (3,558 contexts) (*'M1.2 hasTestingData'* module) and the gold standard (*'M7.1: hasGoldStandard'* module), whilst the SMS data from SemEval 2013 Task 2A (Wilson *et al.*, 2013) contains only testing data (2,175 contexts) (*'M1.2 hasTestingData'* module) and gold standard (*'M7.1: hasGoldStandard'* module).

The gold standard is especially important as it refers to the testing data whose polarity is labelled by human annotators, and is assumed to be correct. This will be used to measure the accuracy of the experiments reported here.

Besides these data, sentiment lexicons were also used, and are described in the following section.

The data format is as follows:

id1<TAB>id2<TAB>start\_token<TAB>end\_token<TAB>unknown<TAB>tweet\_text

For example :

218775148495515649	111114	4	4	unknown	Musical awareness: Great
Big Beautiful Tomorrow has an ending, Now is the time does not					

258965201766998017	111116	17	17	unknown	On Radio786 100.4fm
7:10 Fri Oct 19 Labour analyst Shawn Hattingh: Cosatu's role in the context of unrest in the mining <a href="http://www.radio786.krypton.co.za">http://www.radio786.krypton.co.za</a>					

**Figure 3-3:** Example of Tweet data formats that were received

<sup>38</sup> Combination of Twitter training data and development data; they were obtained from the task's organiser. Source: <https://www.cs.york.ac.uk/semeval-2013/task2/index.html>

The data format is as follow:

id1<TAB>id2<TAB>start\_token<TAB>end\_token<TAB>unknown<TAB>sms\_text

For example :

11350      111118      0      0      unknown      Haha... I want to see. E macdonalds here  
cheaper. Yum yum.

10577      111139      11      11      unknown      After I make u smile I will make u  
angry or cry again. I think I shouldnt talk so much next time. Anyway\u002c u must finish e  
course ok? Btw, how is e project?

**Figure 3-4:** Example of SMS data format that were received

### 3.2.2 Sentiment Lexicons

In addition to the aforementioned datasets, sentiment lexicons were also used. Sentiment lexicons are lexicons (dictionaries) with sentiment values attached to each word, as described in Section 2.1.3.7. Sentiment lexicons are controlled using the '*MI.3 hasSentimentLexicons*' module, which consists of five sentiment lexicons. The '*SR1: hasHuAndLiuLexicon*' module refers to Hu and Liu's lexicons (Hu and Liu, 2004)<sup>39</sup> (6780 words) (HL), which were collected over many years by Hu and Liu, starting in 2004 with their work on online customer product reviews (Hu and Liu, 2004). The '*SR2: hasMPQALexicon*' module refers to the MPQA Subjective Lexicon (MPQA)<sup>40</sup> (8221 words), which was created by Wilson *et al.* (2005b) using a set of approximately four hundred documents. The '*SR3: hasAFINNLexicon*' module refers to the AFINN Lexicon (AFINN)<sup>41</sup> (2477 words), which was created from Twitter between 2009-2011 by Nielsen (2011a) for use in the United Nation Climate Conference (COP15). The '*SR4: hasSentiWordNet Lexicon*' module and '*SR5: hasSentiStrengthLexicon*' module refer to the lexicons from SentiWordNet (SWN) (Baccianella *et al.*, 2010a) and SentiStrength (SS) (Thelwall *et al.*, 2010b), and are described in Section 3.2.5.

<sup>39</sup> <http://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html#lexicon>

<sup>40</sup> [http://mpqa.cs.pitt.edu/lexicons/subj\\_lexicon/](http://mpqa.cs.pitt.edu/lexicons/subj_lexicon/)

<sup>41</sup> [http://www2.imm.dtu.dk/pubdb/views/publication\\_details.php?id=6010](http://www2.imm.dtu.dk/pubdb/views/publication_details.php?id=6010)

### 3.2.3 Combination

Combination is the function for combining training data (*'M1.1: hasTrainingData'* module) and sentiment lexicons (*'M1.3 hasSentimentLexicons'* module), which are controlled using *'C2: Combination'* class. In the combination process, words that duplicate, overlap or contradict in sentiment were removed from the combination process. The idea for removing these words was adapted from previous works (Melville *et al.*, 2009; Yuan *et al.*, 2013; Refaee and Rieser, 2014).

### 3.2.4 Pre-processing

Pre-processing is the process used to control features that can be used in data pre-processing. *'C3: PreProcessing'* class is used to control seven features. The details of each feature are presented below.

Emoticons (Safko, 2010) are symbol combinations used to represent facial expressions in text data such as emails, Tweets, and SMS messages. For example, :-) refers to a smile and :-( refers to a sad face. Emoticons were collected manually from the training datasets and then matched against a well-known collection of emoticons<sup>42</sup> for labelling as positive or negative. After that, these emoticons and labels were stored under *'M3.1 hasLabelEmoticon'* for use in labelling testing data.

Negative verbs have an effect on the polarity; therefore, they were expanded and converted into the full form using the function in *'M3.2 hasConvertNegativeVerb'*. For example, 'don't' was expanded and converted to 'do not'.

In Twitter features, there are three main features that are used as symbols to represent the meaning of usernames, URLs and tags. A Twitter username is a unique name shown in the user's profile and may be used for both authentication and identification. This is shown by prefacing the username with an @ symbol. When a Tweet is directed at an individual or particular entity, this can be shown by including @username. For example, a Tweet directed at 'tawunrat' would include the text '@tawunrat'. Before URLs are posted to Twitter, an online

---

<sup>42</sup> [http://en.wikipedia.org/wiki/List\\_of\\_emoticons](http://en.wikipedia.org/wiki/List_of_emoticons)

service (t.co<sup>43</sup>) is used by Twitter to automatically shorten URLs to a maximum of 22 characters. For example, the URL ‘https://itunes.apple.com/us/album/southpaw-music-from-inspired/id1012586856?linkId=15816529’ would be shortened to ‘apple.co/1IIKaQD’. Hashtags are used to represent keyword and topics on Twitter by using ‘#’ character followed by words or phrases, such as ‘#BangkokThailand’. The function used for removing both username and URL features from the data are in the ‘M3.3 hasRemove@UserURLs’ module. On the other hand, hashtag features are replaced with the word(s) following the # symbol in the data by using the function in ‘M3.4 hasConvert#hashtag’ module. For example, ‘#BangkokThailand’ was replaced by ‘BangkokThailand’.

Repeated letters (Norton *et al.*, 2005) are used for emphasis in the data. They were reduced and replaced using a simple regular expression by two of the same characters from the function in the ‘M3.5 hasReduceRepeatLetter’ module. For example, ‘happpppppy’ will be replaced with ‘happy’. This idea is also used in Bifet and Frank (2010); Raez *et al.* (2012); Muhammad *et al.* (2013); Abdul-Mageed and Diab (2014).

Slang words (Zappavigna, 2012) are composed of information and phrases which are mostly used in short form, such as ‘FYI’. The slang corpora from the noslang dictionary<sup>44</sup> and the function in ‘M3.6 hasConvertSlang’ module were used to convert slang words in the data to their full form. For example, ‘FYI’ was converted to ‘for your information’.

Stopwords (Bird *et al.*, 2009b) are frequently used words. Stopwords have little meaning and are less important, such as ‘a’, ‘an’, ‘the’, ‘there’, ‘those’. The list of English stopwords from Natural Language ToolKit (NLTK) (Bird, 2006b) and the function in ‘M3.7 hasRemoveStopwords’ module were used for matching and removing the words from the data.

Special characters (Norton *et al.*, 2005) are characters or symbols such as [, {, ?, and !. The function in ‘M3.8 hasRemoveSpecialCharacter’ module was used for removing special characters.

---

<sup>43</sup> <https://support.twitter.com/entries/109623>

<sup>44</sup> <http://www.noslang.com/dictionary/>

### 3.2.5 Sentiment Resources

Sentiment resources are resources that automatically extract the sentiment from a phrase or sentence, as described in Section 2.1.3.6. Two sentiment resources are controlled by using ‘*C5: SentimentResources*’ class. They are SentiWordNet (SWN) (Baccianella *et al.*, 2010a) and SentiStrength (Thelwall *et al.*, 2010b).

SentiWordNet (SWN) (Baccianella *et al.*, 2010a) is the result of the automatic annotation of all the synsets of WordNet (Fellbaum, 2010; Princeton University, 2010), described in Section 2.1.3.6. SentiWordNet (SWN) (Baccianella *et al.*, 2010a) is controlled under the ‘*M5.1: hasSentiWordNet*’ module. On the other hand, SentiStrength (Thelwall *et al.*, 2010b) is the sentiment analysis methodology used to judge whether a sentence has a positive or negative sentiment, as described in Section 2.1.3.6. SentiStrength (Thelwall *et al.*, 2010b) is controlled under the ‘*M5.2: hasSentiStrength*’ module.

There are two approaches to using these sentiment resources. The first approach is to train the testing dataset directly into the sentiment resources. For ‘*M5.1: hasSentiWordNet*’ module, Denecke (2008) methods are used. In the methods, the score of each polarity of each synset of the word are combined and divided by using the number of synsets. After that the scores of sentences are generated by summing the scores of each word in the sentences and dividing by the number of those words. This method is also used in Devitt and Ahmad (2007); Thet *et al.* (2009); Sing *et al.* (2012); Guerini *et al.* (2013). An example is given in Figure 3.5.

Conversely, ‘*M5.2: hasSentiStrength*’ used the website of SentiStrength (Thelwall *et al.*, 2010b); the testing dataset was passed into the server using our application in Python (Python Software Foundation, 2001). After that, the accuracy was automatically calculated by the server.

In the second approach, the lexicons from SentiStrength<sup>45</sup> (SS) (Thelwall *et al.*, 2010b) (Section 2.1.3.6) and SentiWordNet<sup>46</sup> (SWN) (Baccianella *et al.*, 2010a) (Section 2.1.3.6) were downloaded and used as the training dataset as were the sentiment lexicons in Section 3.2.2. Their lexicons are represented in the

<sup>45</sup> <http://www.softpedia.com/get/Others/Home-Education/SentiStrength.shtml>

<sup>46</sup> <http://sentiwordnet.isti.cnr.it/download.php>

'SR4: *hasSentiWordNetLexicon*' module and the 'SR5: *hasSentiStrengthLexicon*' module.

For example:

The sentence 'I hate summer.' The result string after pre-processing is 'hate summer'. These two words are passed through in SentiWordNet for corresponding synsets.

For the input term 'hate', SentiWordNet contains 2 synset entries. After summing the scores and dividing by the number of synsets, the results are: positive 0.0625, negative 0.375 and objective 0.5625.

For the input term 'summer', SentiWordNet contains 3 synset entries. After summing the scores and dividing by the number of synsets, the results are: positive 0, negative 1 and objective 0.

The sentences score are summed from the number of each term and divided by the number of terms. The result of this sentence is: positive 0.0313, negative 0.6875 and objective 0.2813.

**Figure 3-5:** Example of sentences that used Denecke (2008) methods

### 3.2.6 Supervised Learning Algorithms

There are two types of supervised learning algorithms in the 'C6: *MachineLearningAlgorithm*' class. They are machine learning algorithms and ensemble learning algorithms.

#### I. Machine learning algorithm

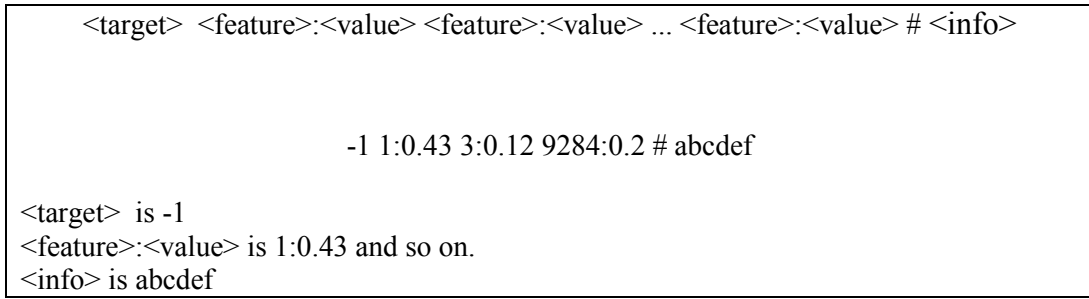
A machine learning algorithm is an algorithm based on input data; it uses that data for making the final decision and prediction, as described in Section 2.2. The three single machine learning algorithms that were used are Naïve Bayes (Tan *et al.*, 2009), Support Vector Machine (SVM) (Kecman, 2005) and Maximum Entropy Modelling (MaxEnt) (Harte, 2011).

The implementation of Naïve Bayes (Tan *et al.*, 2009) (Section 2.2) was used from NLTK (Bird, 2006b). NLTK (Bird, 2006b)<sup>47</sup> is a widely-used machine learning open source platform that was developed using Python (Python Software Foundation, 2001). There is no special format for using Naïve Bayes (Tan *et al.*,

<sup>47</sup> <http://www.nltk.org/>

2009) in NLTK (Bird, 2006b). Naïve Bayes (Tan *et al.*, 2009) is controlled using the ‘M6.1: *hasNaiveBayes*’ module.

For the Support Vector Machine (SVM) approach (Kecman, 2005) (Section 2.2), the implementation, called SVMLight (Joachims, 2002a; Joachims, 2002b), was used. Before applying the data to SVMLight (Joachims, 2002a; Joachims, 2002b), the data needed to be changed to a numerical format, as shown in Figure 3.6. SVM (Kecman, 2005) is controlled using the ‘M6.2: *hasSupportVectorMachine*’ module.



**Figure 3-6:** Example of data format in SVMLight (Joachims, 2002a; Joachims, 2002b)

In this format, ‘target’ represents the polarity of contexts/phrases; ‘feature’ refers to a term in the document, and ‘value’ refers to a feature weight. For a ‘value’, Tf-Idf was used. Tf-Idf is the combination of term frequency (tf) and inverse document frequency (idf), which is a weight value often used in text mining and information retrieval. This weight is a statistical measure for evaluating the relative importance of words in a document within the collection (Manning *et al.*, 2008a). The equation of Tf-idf is defined as (5),

$$tf - idf_{t,d} = tf_{t,d} * idf_t \quad (5)$$

Source: (Manning *et al.*, 2008a)

where  $tf - idf_{t,d}$  is the weighting the scheme assigns to term  $t$  in document  $d$ .

Term frequency ( $tf$ ) is used to measure how frequently the term appears in the document, as in (6).



$$tf_{t,d} = \frac{n_{t,d}}{\sum_k n_{k,d}} \quad (6)$$

Source: (Manning *et al.*, 2008a)

where  $n_{t,d}$  is the number of term  $t$  appears in a document  $d$ .  $\sum_k n_{k,d}$  is the total number of terms  $k$  in the document  $d$ .

Inverse document frequency (*idf*) is used to measure the importance of the term; for example, whether the term is common or rare in the collection, as in (7),

$$idf_t = \log \frac{D}{d_t} \quad (7)$$

Source: (Manning *et al.*, 2008a)

where  $D$  is the total number of documents in the collection in the corpus, and  $d_t$  is the number of documents  $d$  which term  $t$  appears.

The default settings of SVMLight (Joachims, 2002a; Joachims, 2002b) were used throughout. This meant that we used a linear kernel that did not require any parameters.<sup>48</sup>

For Maximum Entropy Modelling (MaxEnt) (Harte, 2011) (Section 2.2), the default setting of the Maximum Entropy Modelling Toolkit from Le (2004) was used. The data format of MaxEnt (Harte, 2011) is the same as SVMLight (Joachims, 2002a; Joachims, 2002b), as mentioned above. MaxEnt (Harte, 2011) is controlled using the '*M6.3: hasMaximumEntropyModelling*' module.

## II. Ensemble learning algorithm

Ensemble learning algorithms are algorithms where multiple classifiers are trained on the same training data and the resulting trained system is used to make the final prediction, as described in Section 2.3. Four ensemble learning algorithms that were used are Majority voting (Polikar, 2012), Stacking (Wolpert, 1992), Arbiter Tree (Chan and Stolfo, 1993) and Combiner Tree (Chan and Stolfo, 1997).

---

<sup>48</sup>Based on default setting of SVMLight

Majority voting (Polikar, 2012) is a basic and simple algorithm that uses the combination of classifiers. Majority voting (Polikar, 2012) is controlled by using the ‘M6.4: *hasMajorityVoting*’ module.

Stacking (Wolpert, 1992) is a technique that uses two classifier levels: base classifier and meta-classifier. Stacking (Wolpert, 1992) is controlled by using the ‘M6.5: *hasStacking*’ module.

Arbiter Tree (Chan and Stolfo, 1993) is a method that uses training data output that has been classified using base classifiers with selection rules. Arbiter Tree (Chan and Stolfo, 1993) is controlled by using the ‘M6.6: *hasArbiterTree*’ module.

Combiner Tree (Chan and Stolfo, 1997) is a method that is trained directly by the training output from base classifiers that have passed the composition rules. Combiner Tree (Chan and Stolfo, 1997) is controlled by using the ‘M6.7: *hasCombinerTree*’ module.

For more details of each method in the ensemble learning algorithms, see Chapter 5.

### 3.2.7 Evaluation Method

The evaluation method is used to measure the accuracy of classification results. The method that is commonly used is known as the F-score or F-measure (Powers, 2011). The process of F-score is controlled by using the ‘M7.1: *hasGoldStandard*’ method.

The F-score method comprises precision and recall. Precision can be defined as the number of data that are correct, while recall is defined as the number of correct data that are generated. Their equations are defined as in (8), (9) and (10):

$$Precision = \frac{TP}{TP + FP} \quad (8)$$

$$Recall = \frac{TP}{TP + FN} \quad (9)$$

$$F - score = 2 * \left( \frac{Precision * Recall}{Precision + Recall} \right) \quad (10)$$

Source: (Witten and Frank, 2005)

TP is a true positive. TP will be defined if the data is in category A and the programme says that the data is in category A. In contrast, if the programme states incorrectly that the data is in category A, this is a false positive (FP). Conversely, if the programme states incorrectly that the data is not in category A, this is a false negative (FN).

### 3.2.8 Data Output

The data output's format is composed of the full original text, part of the words or a phrase from the text and polarity, which is controlled using the function in the '*C8: Output*' class.

The data output format is as follows:

id<TAB>original\_text<TAB>word\_phrase<TAB>polarity

For example:

01 Musical awareness: Great Big Beautiful Tomorrow has an ending, Now is the time does not  
Beautiful        positive

02 On Radio786 100.4fm 7:10 Fri Oct 19 Labour analyst Shawn Hattingh: Cosatu's role in the  
context of unrest in the mining <http://www.radio786.krypton.co.za>  
unrest        negative

**Figure 3-7:** Example of data output from the Tweets

### 3.3 System Operation

The flowchart in Figure 3.8 shows the series of actions for using in the TJP system.

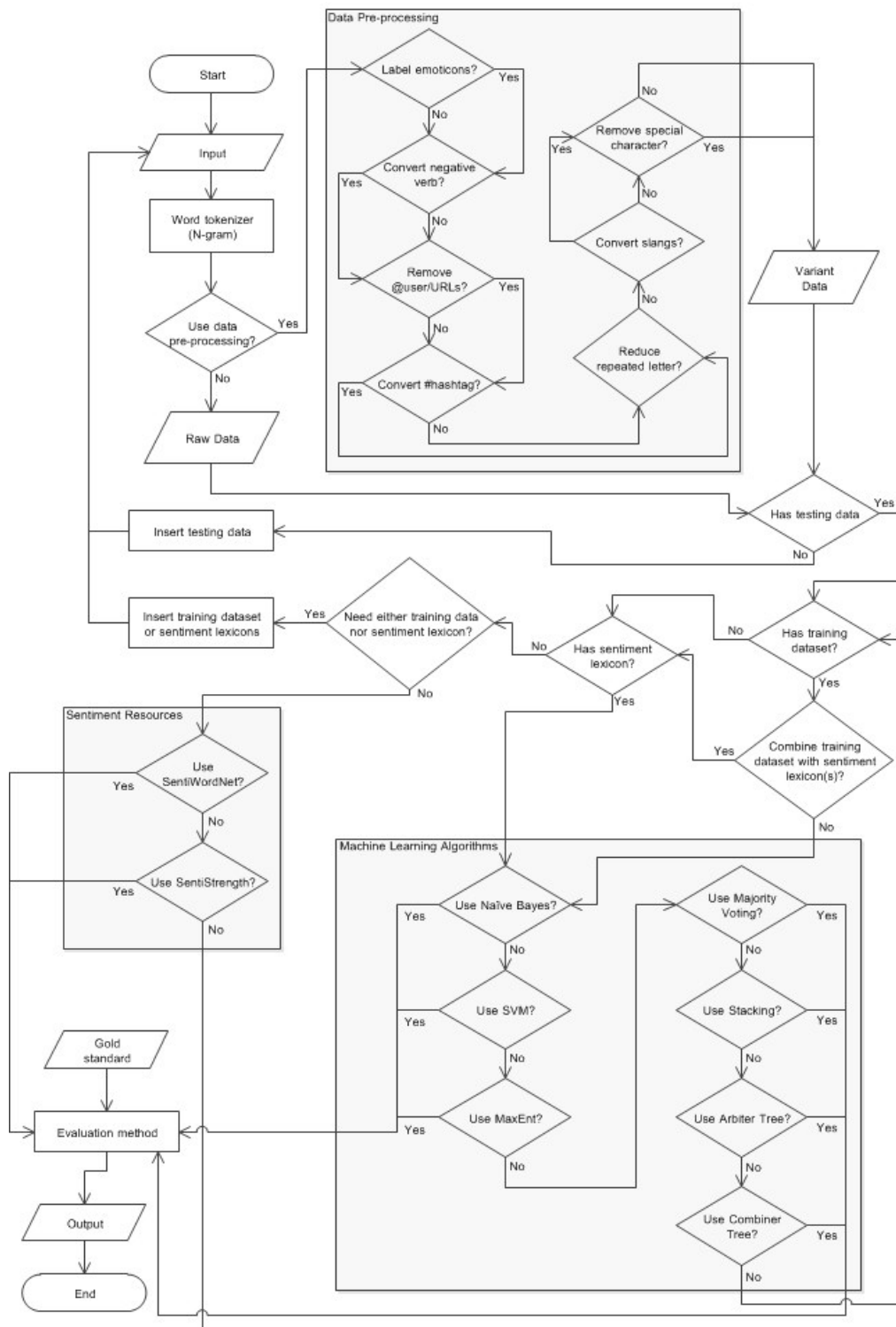


Figure 3-8: The operation of TJP system

### 3.4 Conclusion

This chapter presents the overview of TJP's system using the user interface and class diagram. The main inputs of TJP are from SemEval 2013 Task 2A, which is composed of Twitter and SMS datasets. Twitter datasets contain training data, testing data and gold standard. The SMS datasets contain only testing data and gold standard. Besides these datasets, sentiment lexicons are also used as training data. Moreover, sentiment lexicons are also combined with Twitter training data. In the data pre-processing, there are seven features; these are emoticons, negative verbs, Twitter features, repeated letters, slang words, stopwords and special characters. There are two classifier processes in TJP: sentiment resources and machine learning algorithm. Sentiment resources are SentiWordNet (SWN) (Baccianella *et al.*, 2010a) and SentiStrength (SS) (Thelwall *et al.*, 2010b). There are two approaches in this classifier. The first approach is to train testing data directly to sentiment resources. The second approach is to download and use their lexicon as training data for training with the machine learning algorithm. On the other hand, there are two types of machine learning algorithm that were used: single machine learning algorithm and ensemble learning algorithm. Single machine learning algorithms are composed of Naïve Bayes (Tan *et al.*, 2009), Support Vector Machine (SVM) (Kecman, 2005) and Maximum Entropy Modelling (MaxEnt) (Harte, 2011). Ensemble learning algorithms consist of Majority voting (Polikar, 2012), Stacking (Wolpert, 1992), Arbiter Tree (Chan and Stolfo, 1993) and Combiner Tree (Chan and Stolfo, 1997). After getting the results from both sentiment resources and machine learning algorithms, they were passed through the evaluation method. In the evaluation method, the F-score is used to evaluate the data, which is then presented in a readable format.

Having described the TJP system, the actual experiments for testing the factors involved in sentiment analysis is described in the next chapter.

## Chapter 4 : Factorial Experiments in Sentiment Analysis

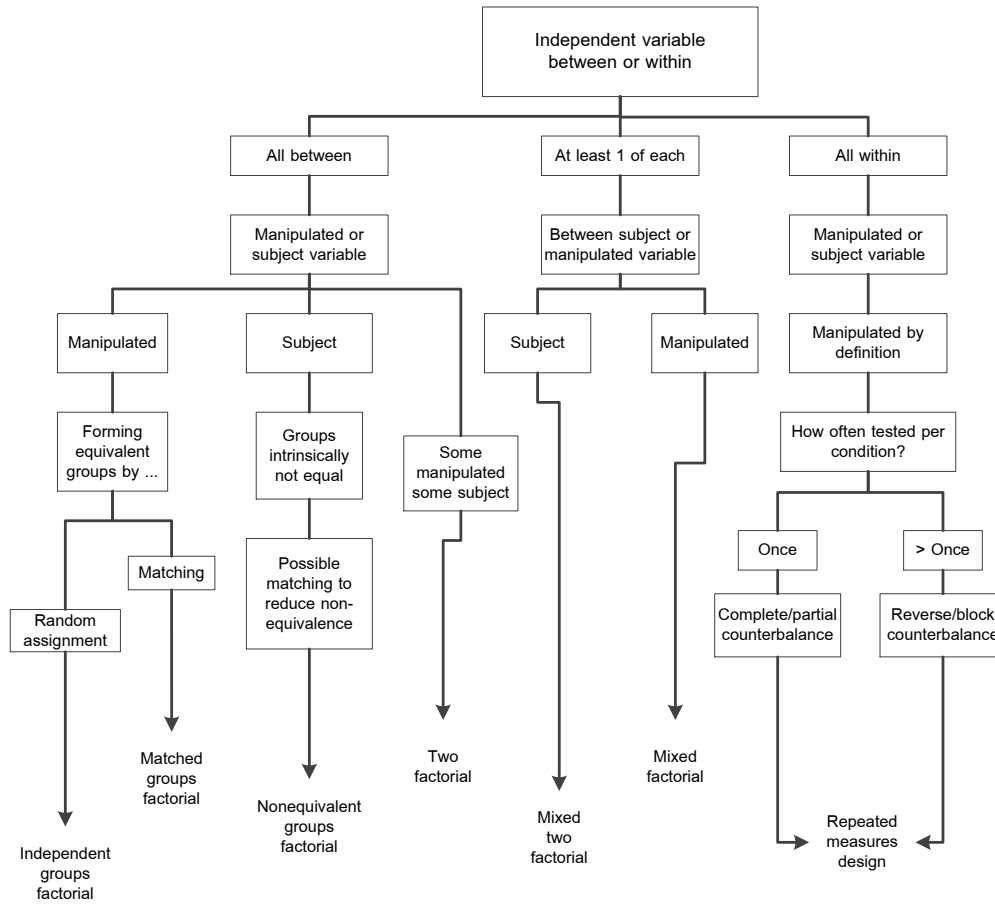
This chapter describes systematic experiments to identify the factors that have an impact on sentiment analysis performance. Such factors may include: sentiment lexicon(s), sentiment resource(s) and machine learning algorithms.

This chapter is organised as follows: Section 4.1 provides an overview of the factorial experiment design, followed by blocking comparative experimental design in Section 4.2. Our experimental design is presented in Section 4.3. The results and analysis are discussed in Section 4.4, followed by the evaluation data and analysis in Section 4.5.

### 4.1 Factorial Experimental Design

Experiments that study the effects of one or more factors are known as factorial experiments. Factorial experimental design is an area of statistics that impacts on experimental disciplines such as psychology or agriculture, where possible combinations of factor levels are investigated (Montgomery, 2013b).

Some standard terms are used in factorial experimental design; such as independent variable, dependent variable, between-subjects, within-subjects, subject variable and manipulated variable. *Independent variable* is a variable that causes the observed results, whilst the *dependent variable* is one that is effected by the independent variable. *Between-subjects* are independent variables in which a different group of subjects is used for each level. *Within-subjects* are independent variables that are manipulated or subject variable by testing each subject at each level. *Manipulated variable* is a variable that can be controlled in the experiment while the *subject variable* cannot be manipulated or controlled. There are designs in factorial experiments as illustrated in Figure 4.1.



**Figure 4-1:** Decision tree diagram of factorial design  
(Goodwin, 2009)

As can be seen in Figure 4, the independent variable can be tested in terms of whether it is between-subjects, within-subject or at least one of each subject. If the independent variable is between-subject and manipulated, the design will be called *independent group design* if equivalent groups are created by simple random assignment. In contrast, if equivalent groups are created using the matching procedure, the design is known as *matched groups design*. Moreover, if the independent variable is tested between subjects as a subject variable, *non-equivalent groups design* will be used. Both a manipulated variable and subject variable are sometimes included in independent variable between-subjects factorials. According to these, the design can yield interaction between two factorial; the design is called *two factorial design*. Conversely, *mixed two factorial design* will be used if the independent design includes both between-subjects and within-subjects and is tested as a subject variable. On the other hand,

if it is tested as manipulated variable, then the *mixed factorial design* is used. Finally, if the independent variable is tested within-subjects, *repeated measures factorial design* is used.

In this study the repeated measures design is appropriate. There are three factors (independent variables) in our experiments; three machine learning algorithms that are declared to be factors (independent variables) in our experiments. The machine learning algorithm is a within-subject variable that is tested as a subject variable. Each subject is tested using each level of the variable, which is a type of training dataset. For the analysis in the factorial experimental design, Chatfield (1983b) suggests using the comparative experimental design. The comparative experimental design can be divided into two groups; simple and blocking comparative experiments. The details of the simple comparative experiment can be found in Appendix III, while the details of the blocking comparative experiment design are described in the following section.

## **4.2 Blocking Comparative Experimental Design**

There are two processes in blocking comparative design that are commonly used in factorial experimental designs (Montgomery, 2013a). They are randomised complete block design (RCBD), and balanced incomplete block design (BIBD). The details of RCBD, which was used, are described in the following section, while the details of BIBD can be found in Appendix IV.

### **4.2.1 Randomised Complete Block Design**

The Randomised Complete Block Design (RCBD) is an extension of the paired t-test (dependent t-test) which may be used where the factor of interest has more than two levels; that is, more than two treatments must be compared (Montgomery and Runger, 2007). The general procedure for the randomised complete block experiment consists of selecting a block and running a complete replica of the experiment in each block. The data records in RCBD are presented in Table 4.1. For Analysis of Variance (ANOVA)<sup>49</sup> for RCBD, the forms shown

---

<sup>49</sup> Statistical method used for analyse the differences between more than two population means



in Table 4.2 were used (Chatfield, 1983a; Montgomery and Runger, 2007; Montgomery, 2013a).

Treatments (Methods)	Block			
	1	2	3	4
1	$y_{11}$	$y_{12}$	$y_{13}$	$y_{14}$
2	$y_{21}$	$y_{22}$	$y_{23}$	$y_{24}$
3	$y_{31}$	$y_{32}$	$y_{33}$	$y_{34}$

**Table 4-1:** Data records of RCBD

Source of variation	Sum of squares	Degrees of freedom	Mean square	$F_0$
<b>Treatments (Methods)</b>	$SS_{Treatment}$	$a - 1$	$\frac{SS_{Treatments}}{a - 1}$	$\frac{MS_{Treatments}}{MS_E}$
<b>Blocks</b>	$SS_{Blocks}$	$b - 1$	$\frac{SS_{Blocks}}{b - 1}$	
<b>Error</b>	$SS_E$	$(a - 1)(b - 1)$	$\frac{SS_E}{(a - 1)(b - 1)}$	
<b>Total</b>	$SS_T$	$N - 1$		

**Table 4-2:** ANOVA for RCBD  
(Chatfield, 1983a; Montgomery and Runger, 2007; Montgomery, 2013a)

The formulae for the sum of squares in ANOVA for RCBD are

$$SS_{Treatment} = \frac{1}{b} \sum_{i=1}^a y_{i.}^2 - \frac{y_{...}^2}{ab} \quad (11)$$

$$SS_{Blocks} = \frac{1}{a} \sum_{j=1}^b y_{.j}^2 - \frac{y_{...}^2}{ab} \quad (12)$$

$$SS_E = SS_T - SS_{Treatments} - SS_{Blocks} \quad (13)$$

$$SS_T = \sum_{i=1}^a \sum_{j=1}^b y_{ij}^2 - \frac{y_{...}^2}{ab} \quad (14)$$

Source: (Chatfield, 1983a; Montgomery and Runger, 2007; Montgomery, 2013a)

For computing the degrees of freedom which is the number of intensive variables in the system (Roy, 2002; Kushwaha, 2009), in ANOVA for RCBD,  $a$  refers to the number of treatments;  $b$  refers to the number of blocks,  $y_{ij}$  refers to

the values when method  $i$  is used on block  $j$ .  $y_{i.}$  is the total of all observations taken under method  $i$ ,  $y_{.j}$  is the total of all observations in block  $j$ ,  $y_{...}$  is the grand total of all observations, and  $N$  is the total number of observations which equal  $ab$ .  $F_0$  is used for testing the null hypothesis that the effects of the treatment are all zero.

### 4.3 Experimental Design

In this section, the following experimental design is presented; ‘*the effect of machine learning on system performance in sentiment analysis by using types of treatments*’, with the TJP system that was used (Chapter 3).

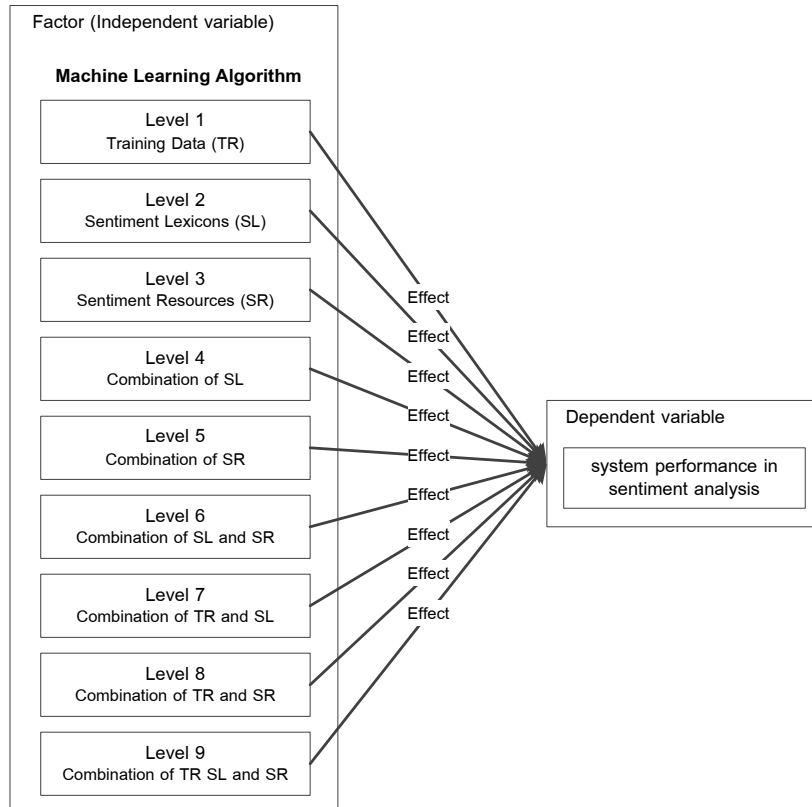
The types of treatments refer to the data that were used as training data for machine learning; training dataset (TR) (Section 3.2.1), sentiment lexicons (SL) (Section 3.2.2), sentiment resources (SR) (Section 3.2.5) and a combination of these.

Moreover, three machine learning algorithms were used. These were Naïve Bayes (Tan *et al.*, 2009) (Section 2.2), Support Vector Machine (SVM) (Kecman, 2005) (Section 2.2) and Maximum Entropy Modelling (MaxEnt) (Harte, 2011) (Section 2.2).

In addition, as explained in Section 4.1, the repeated measures design is appropriate in this case. Accordingly, each machine learning algorithm (within-subject) was tested using each level of the variable (independent variable), which is a type of training data. After that, the results were compared. The types of training data (treatment) are illustrated in Figure 4.2. Moreover, the RCBD blocking outlined in Section 4.2.1 was selected. The reason for this is that, the blocks in the experiment could be filled without missing any treatments, as presented in Table 4.3.

However, before starting the factorial experiment, the features in data pre-processing (Section 3.5) were investigated to decide which features should be used. In the investigation, each feature was combined and tested using Naïve Bayes (Tan *et al.*, 2009) (Section 2.2). The flowcharts of the combination of features are illustrated in Figure 4.2. The results reveal that using the combination of all features achieved better accuracy and had a considerable effect on system

performance, as shown in Table 4.4. Therefore, the output of this combination was transferred and converted for use with the other machine learning algorithms: Support Vector Machine (Kecman, 2005) (Section 2.2) and Maximum Entropy Modelling (Harte, 2011) (Section 2.2). The evaluation method that was used is mentioned in Section 3.7. The results of the factorial experiment are presented in the following section.



**Figure 4-2:** Declare factors and response variable

Factors	Levels (treatments)									
	Training data	Sentiment lexicons (SL)	Sentiment resources (SR)	Combination of SL	Combination of SR	Combination of SL and SR	Combination of TR and SL	Combination of TR and SR	Combination of TR, SR and SL	
Naïve Bayes (NB)	✓	✓	✓	✓	✓	✓	✓	✓	✓	
Support Vector Machine (SVM)	✓	✓	✓	✓	✓	✓	✓	✓	✓	
Maximum Entropy Modelling (MaxEnt)	✓	✓	✓	✓	✓	✓	✓	✓	✓	

**Table 4-3:** Example of RCBD data recorded in experiment

Testing data Training data	RAW	Emoticon	Negation	@user, URL	Hashtag	Repeated letters	Slangs	Stopwords	Special characters
<b>RAW</b>	58.74	58.74	50.03	50.05	50.05	50.05	50.09	48.53	48.32
<b>Emoticon</b>	75.47	75.74	78.68	78.57	78.57	78.57	78.21	77.86	77.58
<b>Negation</b>	71.05	71.42	79.87	79.87	79.87	79.87	79.53	79.78	80.06
<b>@user, URL</b>	71.05	71.42	79.87	79.87	79.87	79.87	79.53	79.78	80.06
<b>Hashtag</b>	73.59	73.93	80.21	80.21	80.21	80.21	79.87	80.18	80.26
<b>Repeated letters</b>	73.70	74.04	80.31	80.31	80.31	80.38	80.00	80.31	80.39
<b>Slangs</b>	73.93	74.25	80.06	80.06	80.06	80.13	80.39	80.32	80.54
<b>Stopwords</b>	73.64	73.69	80.09	80.09	80.09	80.15	80.38	80.38	80.59
<b>Special characters</b>	76.11	76.41	80.55	80.55	80.55	80.62	80.81	80.81	<b>81.06</b>

**Table 4-4:** The results of each feature analysed by using Naïve Bayes (F-score)

#### 4.4 Results and Analysis

In the following (Table 4.5 and Figure 4.3), the results of the factorial experimental design using the Twitter dataset (Section 3.2.1) are presented.

		1 NB	2 SVM	3 MaxEnt
TR	1 TR	81.06	82.62	59.93
	2 HL	38.48	55.77	42.94
SL	3 MPQA	34.27	57.15	32.18
	4 AFINN	33.62	69.56	31.85
SR	5 SWN	62.79	59.70	33.30
	6 SS	29.46	37.64	30.81
Combination of SL	7 HL + MPQA	40.72	64.38	32.69
	8 HL + MPQA + AFINN	41.38	69.47	33.25
	9 HL + AFINN	37.43	61.82	32.31
	10 MPQA + AFINN	40.22	65.27	33.04
Combination of SR	11 SWN + SS	64.17	62.84	33.78
	12 SS + HL	30.83	53.56	31.66
Combination of SL and SR	13 SS + HL + MPQA	38.75	58.96	32.82
	14 SS + HL + MPQA + AFINN	38.07	62.02	33.21
	15 SS + HL + AFINN	35.58	55.98	32.31
	16 SS + MPQA	33.86	55.46	32.55
	17 SS + MPQA + AFINN	36.81	58.74	33.08
	18 SS + AFINN	30.62	47.12	31.99
	19 SWN + HL	61.86	60.34	33.70
	20 SWN + HL + MPQA	61.03	60.60	33.74
	21 SWN + HL + MPQA + AFINN	64.66	62.93	34.29
	22 SWN + HL + AFINN	63.08	62.73	34.29
	23 SWN + MPQA	61.25	61.00	33.59
	24 SWN + MPQA + AFINN	63.66	62.02	34.33
	25 SWN + AFINN	62.56	61.11	34.27
	26 SWN + SS + HL	61.99	62.44	33.87
	27 SWN + SS + HL + MPQA	62.79	62.47	33.91
	28 SWN + SS + HL + MPQA + AFINN	65.36	63.18	34.29
	29 SWN + SS + HL + AFINN	63.48	63.15	34.29
	30 SWN + SS + MPQA	61.79	63.21	33.72
	31 SWN + SS + MPQA + AFINN	62.92	62.50	34.37
	32 SWN + SS + AFINN	63.87	61.98	34.31
Combination of TR and SL	33 TR + HL	80.84	82.47	46.88
	34 TR + HL + MPQA	81.26	82.81	46.93
	35 TR + HL + MPQA + AFINN	81.94	83.55	46.99
	36 TR + HL + AFINN	81.74	83.32	47.01
	37 TR + MPQA	82.57	81.99	46.88
	38 TR + MPQA + AFINN	81.73	83.20	46.93
Combination of TR and SR	39 TR + AFINN	82.91	83.00	46.92
	40 TR + SS	79.99	80.46	46.70
	41 TR + SWN	79.49	81.51	46.93
Combination of TR, SL and SR	42 TR + SWN + SS	80.37	81.74	46.99
	43 TR + SS + HL	80.75	81.75	46.90
	44 TR + SS + HL + MPQA	81.36	82.93	46.93
	45 TR + SS + HL + MPQA + AFINN	81.84	83.33	46.99
	46 TR + SS + HL + AFINN	81.47	82.43	47.01
	47 TR + SS + MPQA	80.83	82.34	46.90
	48 TR + SS + MPQA + AFINN	81.57	83.09	46.96
	49 TR + SS + AFINN	81.31	81.72	46.91
	50 TR + SWN + HL	73.33	82.00	47.09
	51 TR + SWN + HL + MPQA	80.91	82.04	47.04
	52 TR + SWN + HL + MPQA + AFINN	81.55	82.82	47.10
	53 TR + SWN + HL + AFINN	81.58	82.96	47.17
	54 TR + SWN + MPQA	80.36	81.15	47.02
	55 TR + SWN + MPQA + AFINN	81.47	82.33	47.08
	56 TR + SWN + AFINN	81.14	82.66	47.08
	57 TR + SWN + SS + HL	81.26	82.28	47.09
	58 TR + SWN + SS + HL + MPQA	81.10	82.19	47.04
	59 TR + SWN + SS + HL + MPQA + AFINN	81.26	82.75	47.10
	60 TR + SWN + SS + HL + AFINN	81.71	82.67	47.17
	61 TR + SWN + SS + MPQA	80.76	81.85	47.04
	62 TR + SWN + SS + MPQA + AFINN	81.49	82.49	47.10
	63 TR + SWN + SS + AFINN	81.43	82.49	47.10

**Table 4-5:** The results from the Twitter dataset

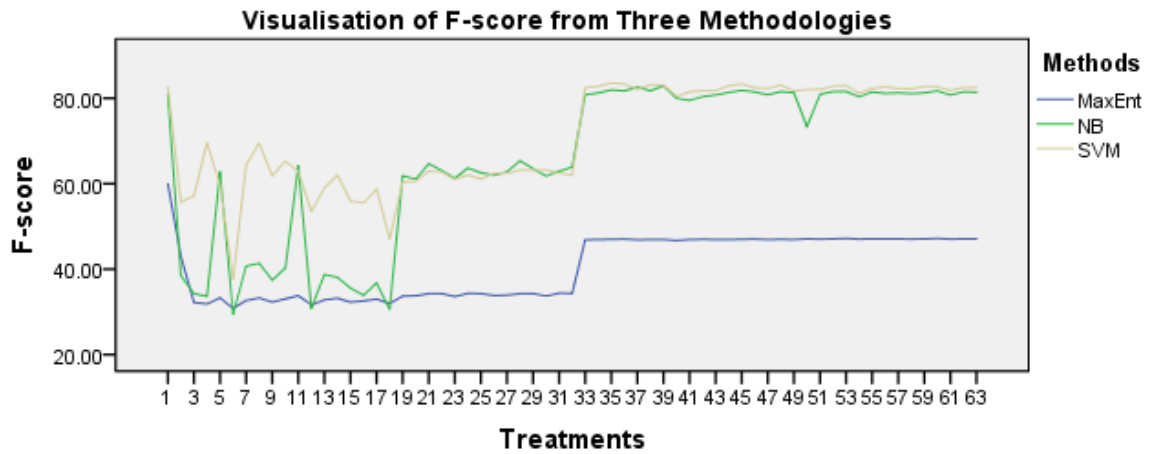


Figure 4-3: Visualisation of F-score from Twitter dataset

#### 4.4.1 Basic Analysis

From the obtained results (Table 4.5), only the results from the training dataset (Section 3.2.1) were used as the baseline results. They achieved F-scores of 80.60%, 82.62% and 59.93% from Naïve Bayes (Tan *et al.*, 2009), SVM (Kecman, 2005) and MaxEnt (Harte, 2011), respectively.

The sentiment lexicons in Section 3.2.2 were used as training data and trained with three machine learning algorithms. The results from AFINN achieved higher level of accuracy than the other sentiment lexicons; HL and MPQA at F-score of 69.56% using SVM (Kecman, 2005). After that, sentiment lexicon was combined together and used as training data. The combination of three sentiment lexicons (HL + MPQA + AFINN) achieved F-score of 69.47% using SVM (Kecman, 2005). However, this result was still lower than when using either single sentiment lexicons or baseline results.

As mentioned in Section 3.2.5, there are two approaches for sentiment resources. The first approach of training the data directly to the sentiment resources achieved F-scores of 78.37% and 72.99% from SentiStrength (Thelwall *et al.*, 2010b) and SentiWordNet (Baccianella *et al.*, 2010a), respectively. Using the second approach, which used the lexicons as training data, the best results of SentiStrength (Thelwall *et al.*, 2010b) achieved at F-score of 37.64% using SVM (Kecman, 2005). On the other hand, the best results of SentiWordNet (Baccianella *et al.*, 2010a) achieved 62.79% using Naïve Bayes (Tan *et al.*, 2009). Furthermore, their lexicons were combined and used as training data; the results

from using Naïve Bayes (Tan *et al.*, 2009) revealed F-score of 64.19% which is better than SVM (Kecman, 2005) and MaxEnt (Harte, 2011).

In addition, sentiment lexicons (Section 3.2.2) and the lexicons of sentiment resources were combined together for use as training data. The combination of SWN + SS + HL + MPQA + AFINN achieved better accuracy than the other combination with F-score of 65.36% from using Naïve Bayes (Tan *et al.*, 2009). However, this combination still achieved a lower degree of accuracy than either combination of sentiment lexicons or single sentiment lexicons or the baseline.

Furthermore, sentiment lexicons (Section 3.2.2) were combined with the training dataset (Section 3.2.1) before being applied to machine learning algorithms. The combination of TR + HL + MPQA + AFINN achieved better accuracy than the others with F-score of 83.55% using SVM (Kecman, 2005). This performance was shown to be superior when compared with the others.

After that, the training dataset was combined with the lexicons of sentiment resources; the combination of TR + SWN + SS achieved a high level of accuracy with F-score of 81.74% using SVM (Kecman, 2005). This performance achieved lower than the baseline results from SVM (Kecman, 2005) but higher than the baseline from Naïve Bayes (Tan *et al.*, 2009) and MaxEnt (Harte, 2011).

Moreover, the combination of the training dataset, sentiment lexicons and lexicons of sentiment resources were used. A high degree of accuracy was achieved from the combination of TR + SS + HL + MPQA + AFINN with F-score of 83.33% but it was still lower than the results from the combination of TR + HL + MPQA + AFINN.

Overall, the results from the combination of sentiment lexicons and lexicons from sentiment resources did not show the improvement in accuracy. Conversely, the results of the combination of training data and sentiment lexicons or lexicons of sentiment resources demonstrated an improvement in accuracy, and also F-score accuracy greater than the baseline. The highest level of accuracy was achieved by the combination of training data and all sentiment lexicons with F-score of 83.55% using SVM (Kecman, 2005). However, in the absence of training datasets, the results from the sentiment lexicon, AFINN achieved a higher degree of accuracy than the others with F-score of 69.56% using SVM (Kecman, 2005).

#### 4.4.2 ANOVA Analysis

For using ANOVA (sample of data entry, see Appendix VII), the hypotheses were set and SPSS (IBM, 2010) was used with the significance level at  $\alpha = 0.05$ .

1.  $H_0$ : The difference machine learning algorithms have no effect on system performance relative to Naïve Bayes<sup>50</sup>  
 $H_1$ : Difference machine learning algorithms have effect on system performance relative to Naïve Bayes
2.  $H_0$ : All machine learning algorithms achieved an equal level of system performance  
 $H_1$ : All machine learning algorithm did not achieve an equal level of system performance

Tests of Factors and Treatments Effects					
Dependent Variable: F-score					
Source	Type III Sum of Squares	df	Mean Square	F	Sig.
Model	721463.698 <sup>a</sup>	65	11099.442	208.520	.000
Treatments	27689.683	62	446.608	8.390	.000
Factor	33578.552	2	16789.276	315.412	.000
Error	6600.483	124	53.230		
Total	728064.181	189			
a. R Squared = .991 (Adjusted R Squared = .986)					

**Table 4-6:** Tests of Factors and Treatments Effects of ANOVA of Twitter dataset

Multiple Comparisons						
Dependent Variable: F-score						
Tukey HSD						
(I) Factor	(J) Factor	Mean Difference (I-J)	Std. Error	Sig.	95% Confidence Interval Lower Bound	Upper Bound
MaxEnt(3)	NB(1)	-24.6533*	1.29994	.000	-27.7370	-21.5696
	SVM(2)	-30.8644*	1.29994	.000	-33.9481	-27.7808
NB(1)	MaxEnt(3)	24.6533*	1.29994	.000	21.5696	27.7370
	SVM(2)	-6.2111*	1.29994	.000	-9.2948	-3.1274
SVM(2)	MaxEnt(3)	30.8644*	1.29994	.000	27.7808	33.9481
	NB(1)	6.2111*	1.29994	.000	3.1274	9.2948
Based on observed means.						
The error term is Mean Square (Error) = 53.228.						
*. The mean difference is significant at the .05 level.						

**Table 4-7:** Multiple comparison analysis of ANOVA of Twitter dataset

<sup>50</sup> Naïve Bayes was set as the baseline in the comparison.



Correlations				
		Treatments	Factors	F-score
Treatments	Pearson Correlation	1	.000	.518**
	Sig. (2-tailed)		1.000	.000
	N	189	189	189
Factors	Pearson Correlation	.000	1	-.531**
	Sig. (2-tailed)	1.000		.000
	N	189	189	189
F-score	Pearson Correlation	.518**	-.531**	1
	Sig. (2-tailed)	.000	.000	
	N	189	189	189

\*\* . Correlation is significant at the 0.01 level (2-tailed).

**Table 4-8:** Correlation analysis of ANOVA of Twitter dataset

From the ANOVA output shown in Table 4.6, ‘Factors’ refer to the machine learning algorithms, while ‘treatments’ refer to the training data that was used. The ‘Factors’ row showed Sig.  $\approx 0.000$ , which is lower than  $\alpha$ , so  $H_0$  is rejected. From this, it can be concluded that different machine learning algorithms affect sentiment analysis performance and all machine learning algorithms did not achieve the same level of system performance.

Since  $H_0$  is rejected, the multiple comparison analysis will be used to determine which ‘Factors’ are similar and which are different, as presented in Table 4.7. The output showed that, we can be 95% confident that all factors are different with SVM (Kecman, 2005) yielding the highest mean F-score while MaxEnt (Harte, 2011) yielded the lowest.

Moreover, correlation analysis was used to discover the relationship between ‘Factors’, ‘Treatments’ and ‘F-score’, as presented in Table 4.8. The results from the Pearson Correlation (Section 2.1.2) reveal that ‘Factors’ is negatively<sup>51</sup> correlated to the ‘F-score’ with a coefficient of 0.531. ‘Treatments’ is positively<sup>52</sup> correlated to the ‘F-score’ with a coefficient of 0.518. Both correlations are significant at a level lower than 0.01. In contrast, there is a coefficient of 0 between ‘Factors’ and ‘Treatments’. In other words, the type of training data does not have any significant impact on the machine learning algorithms.

<sup>51</sup> Negative correlation can be defined as the relationship between two variables in which one variable increases as the other decreases and vice versa.

<sup>52</sup> Positive correlation can be defined as the relationship between two variables in which both variables increase and decrease together.

#### 4.5 Evaluation of SMS Data and Analysis

The experimental design outlined in Section 4.3 was also applied to the SMS dataset (Section 3.2.1). The results are presented below (Table 4.9 and Figure 4.4).

		1 NB	2 SVM	3 MaxEnt
TR	1 TR	85.49	85.05	50.80
SL	2 HL	48.22	58.36	34.20
	3 MPQA	42.33	63.62	39.81
SR	4 AFINN	38.94	74.96	39.94
	5 SWN	65.84	64.28	40.12
Combination of SL	6 SS	38.27	42.71	39.53
	7 HL + MPQA	48.73	66.39	39.99
	8 HL + MPQA + AFINN	45.65	67.89	40.52
	9 HL + AFINN	44.19	65.12	40.22
Combination of SR	10 MPQA + AFINN	45.16	65.62	40.33
	11 SWN + SS	77.09	77.06	39.97
Combination of SL and SR	12 SS + HL	41.05	53.06	40.36
	13 SS + HL + MPQA	49.25	60.57	40.61
	14 SS + HL + MPQA + AFINN	45.06	58.25	40.22
	15 SS + HL + AFINN	42.92	55.43	40.18
	16 SS + MPQA	42.24	56.77	40.43
	17 SS + MPQA + AFINN	42.24	58.91	40.13
	18 SS + AFINN	37.10	48.31	39.97
	19 SWN + HL	73.56	71.10	41.63
	20 SWN + HL + MPQA	76.71	73.25	41.72
	21 SWN + HL + MPQA + AFINN	80.07	76.48	42.16
	22 SWN + HL + AFINN	80.89	78.27	42.24
	23 SWN + MPQA	79.04	73.22	41.64
	24 SWN + MPQA + AFINN	79.04	73.70	42.07
	25 SWN + AFINN	80.64	78.98	42.15
	26 SWN + SS + HL	77.19	76.53	41.86
	27 SWN + SS + HL + MPQA	78.24	76.71	42.09
	28 SWN + SS + HL + MPQA + AFINN	78.93	77.68	42.10
	29 SWN + SS + HL + AFINN	79.45	78.82	42.24
	30 SWN + SS + MPQA	76.99	77.25	42.24
	31 SWN + SS + MPQA + AFINN	78.41	74.56	42.01
	32 SWN + SS + AFINN	79.48	77.42	42.16
Combination of TR and SL	33 TR + HL	84.51	85.54	48.90
	34 TR + HL + MPQA	84.56	85.45	48.83
	35 TR + HL + MPQA + AFINN	85.03	85.78	48.90
	36 TR + HL + AFINN	84.98	85.96	49.00
	37 TR + MPQA	87.85	85.63	48.76
	38 TR + MPQA + AFINN	84.84	86.05	48.83
Combination of TR and SR	39 TR + AFINN	87.25	84.95	48.83
	40 TR + SS	83.72	84.96	48.86
	41 TR + SWN	83.79	84.13	48.83
Combination of TR, SL and SR	42 TR + SWN + SS	84.24	84.99	48.83
	43 TR + SS + HL	84.56	84.98	48.93
	44 TR + SS + HL + MPQA	84.61	85.12	48.83
	45 TR + SS + HL + MPQA + AFINN	84.98	85.36	48.90
	46 TR + SS + HL + AFINN	84.84	85.31	49.00
	47 TR + SS + MPQA	84.61	85.85	48.76
	48 TR + SS + MPQA + AFINN	84.89	86.09	48.83
	49 TR + SS + AFINN	84.14	85.71	48.93
	50 TR + SWN + HL	84.43	84.36	48.90
	51 TR + SWN + HL + MPQA	84.47	84.17	48.83
	52 TR + SWN + HL + MPQA + AFINN	84.66	84.49	48.83
	53 TR + SWN + HL + AFINN	84.57	84.73	48.90
	54 TR + SWN + MPQA	84.33	84.44	48.76
	55 TR + SWN + MPQA + AFINN	84.52	84.81	48.76
	56 TR + SWN + AFINN	84.06	84.45	48.83
	57 TR + SWN + SS + HL	84.52	84.99	48.90
	58 TR + SWN + SS + HL + MPQA	84.38	84.49	48.83
	59 TR + SWN + SS + HL + MPQA + AFINN	84.47	84.76	48.83
	60 TR + SWN + SS + HL + AFINN	84.56	85.08	48.90
	61 TR + SWN + SS + MPQA	84.42	85.08	48.76
	62 TR + SWN + SS + MPQA + AFINN	84.47	85.22	48.76
	63 TR + SWN + SS + AFINN	84.29	85.22	48.83

**Table 4-9:** The results from SMS dataset

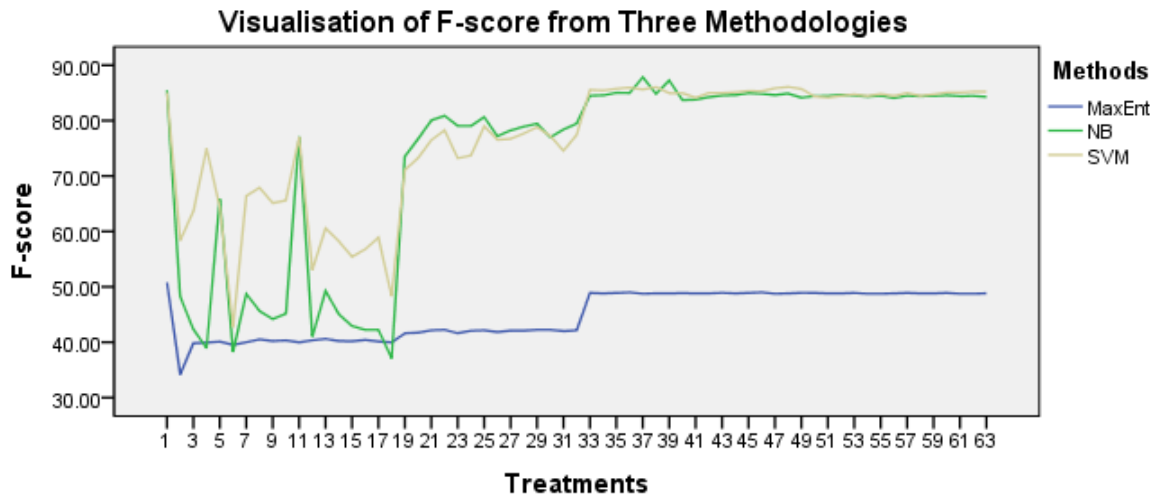


Figure 4-4: Visualisation of F-score from SMS dataset

#### 4.5.1 Basic Analysis

From the obtained results in Table 4.9 and Figure 4.4, the baseline results were obtained from using only the training dataset (Section 3.2.1). They achieved F-scores 85.49%, 85.05% and 50.80% using Naïve Bayes (Tan *et al.*, 2009), SVM (Kecman, 2005) and MaxEnt (Harte, 2011), respectively.

The sentiment lexicons in Section 3.2.2 were used as training data and trained with three machine learning algorithms. The results from AFINN achieved better accuracy performance than the other sentiment lexicons; HL and MPQA at F-score 74.96% using SVM (Kecman, 2005). After that, the sentiment lexicons were combined together and used as training data. The combination of three sentiment lexicons (HL + MPQA + AFINN) achieved an F-score of 67.89% using SVM (Kecman, 2005). However, this performance was still lower than when using either single sentiment lexicons or the baseline.

As mentioned in Section 3.2.5, there are two approaches to sentiment resources. The first approach is to train the data directly to the sentiment resources, which achieved F-scores of 79.83% and 78.85% from SentiStrength (Thelwall *et al.*, 2010b) and SentiWordNet (Baccianella *et al.*, 2010a), respectively. In the second approach the lexicons are used as training data; the best results of SentiStrength (Thelwall *et al.*, 2010b) achieved an F-score of 42.71% using SVM (Kecman, 2005). On the other hand, the best results of SentiWordNet (Baccianella *et al.*, 2010a) achieved an F-score of 65.84% using

Naïve Bayes (Tan *et al.*, 2009). Furthermore, the lexicons were combined and used as training data; the results using Naïve Bayes (Tan *et al.*, 2009) revealed an F-score of 77.09%, which is better than SVM (Kecman, 2005) and MaxEnt (Harte, 2011).

In addition, sentiment lexicons (Section 3.2.2) and the lexicons of sentiment resources were combined together for use as training data. The combination of SWN + HL + AFINN achieved a higher level of accuracy than the other combinations with F-score of 80.89% using Naïve Bayes (Tan *et al.*, 2009). However, this combination still achieved a lower degree of accuracy than the baseline.

Furthermore, sentiment lexicons (Section 3.2.2) were combined with the training dataset (Section 3.2.1) before being applied to the machine learning algorithms. The combination of TR + MPQA achieved a higher level of accuracy than the others with F-score of 87.85% using Naïve Bayes (Tan *et al.*, 2009). This performance was also revealed the best when compared with the others and the baseline.

After that, the training dataset was combined with lexicons of sentiment resources; the combination of TR + SWN + SS achieved an F-score of 84.99% using SVM (Kecman, 2005). However, this performance was still lower than the baseline.

Moreover, the combination of training dataset, sentiment lexicons and lexicons of sentiment resources was used. A high level of accuracy was achieved from the combination of TR + SS + MPQA + AFINN with an F-score of 86.09% using SVM (Kecman, 2005) but this was still lower than the results from the combination of TR + MPQA.

Overall, as in Section 4.4.1, the results revealed that, the combination of sentiment lexicons and lexicons from sentiment resources did not show much improvement in accuracy performance. Conversely, the results of the combination of training data and sentiment lexicons demonstrated improved accuracy performances, with greater F-score accuracy than the baseline. In contrast, the results of the combination of training data and lexicons from sentiment resources did not improve when compared with the baseline. However, the highest degree of

accuracy was achieved using the combination of training data and MPQA with an F-score of 87.85% from Naïve Bayes (Tan *et al.*, 2009). However, in the absence of training datasets, the results from the sentiment lexicon AFINN achieved a greater level of accuracy than the others with F-score of 69.56% using SVM (Kecman, 2005).

#### 4.5.2 ANOVA Analysis

SPSS (IBM, 2010) was used with the same hypotheses as in Section 4.5.2.

Tests of Factors and Treatments Effects					
Dependent Variable: Fscore					
Source	Type III Sum of Squares	df	Mean Square	F	Sig.
Model	853399.312 <sup>a</sup>	65	13129.220	241.114	.000
Treatments	20367.383	62	328.506	6.033	.000
Factor	37954.700	2	18977.350	348.513	.000
Error	6752.083	124	54.452		
Total	860151.395	189			
a. R Squared = .992 (Adjusted R Squared = .988)					

**Table 4-10:** Tests of Factors and Treatments Effects Two-ways ANOVA of SMS dataset

Multiple Comparisons						
Dependent Variable: F-score						
Tukey HSD						
(I) Factor	(J) Factor	Mean Difference (I-J)	Std. Error	Sig.	95% Confidence Interval	
MaxEnt(3)	NB(1)	-28.1579*	1.31478	.000	-31.2768	-25.0390
	SVM(2)	-31.6583*	1.31478	.000	-34.7772	-28.5394
NB(1)	MaxEnt(3)	28.1579*	1.31478	.000	25.0390	31.2768
	SVM(2)	-3.5003*	1.31478	.024	-6.6192	-.3814
SVM(2)	MaxEnt(3)	31.6583*	1.31478	.000	28.5394	34.7772
	NB(1)	3.5003*	1.31478	.024	.3814	6.6192
Based on observed means.						
The error term is Mean Square (Error) = 54.452.						
*. The mean difference is significant at the 0.05 level.						

**Table 4-11:** Multiple Comparison analysis of ANOVA of SMS dataset

Correlations				
		Treatments	Factors	F-score
Treatments	Pearson Correlation	1	.000	.442**
	Sig. (2-tailed)		1.000	.000
	N	189	189	189
Factors	Pearson Correlation	.000	1	-.620**
	Sig. (2-tailed)	1.000		.000
	N	189	189	189
F-score	Pearson Correlation	.442**	-.620**	1
	Sig. (2-tailed)	.000	.000	
	N	189	189	189

\*\* . Correlation is significant at the 0.01 level (2-tailed).

**Table 4-12:** Correlation analysis of ANOVA of SMS dataset

From the ANOVA output (Table 4.10), the ‘Factors’ (machine learning algorithms) row showed Sig.  $\approx 0.000$ , which is lower than  $\alpha$ ; thus,  $H_0$  is rejected, as explained in Section 4.5.2. It can be concluded that different machine learning algorithms affect sentiment analysis performance and not all machine learning algorithms achieved the same level of system performance.

Since  $H_0$  is rejected, the multiple comparison analysis will be used to compare the ‘Factors’, as shown in Table 4.11. The output showed that we can be 95% confident that all the factors are different, with SVM (Kecman, 2005) yielding the highest mean F-score while MaxEnt (Harte, 2011) yielded the lowest, as explained in Section 4.4.2.

Moreover, correlation analysis was used to discover the relationship between ‘Factors’, ‘Treatments’ and ‘F-score’, as presented in Table 4.12. The results from the Pearson Correlation (Section 2.1.2) reveal that ‘Factors’ is negatively<sup>53</sup> correlated to the ‘F-score’ with a coefficient of 0.620. ‘Treatments’ is positively<sup>54</sup> correlated to the ‘F-score’ with a coefficient of 0.442. Both correlations have a significance level lower than 0.01. In contrast, there is no coefficient correlation between ‘Factors’ and ‘Treatments’ due to the significance value of 1. In other words, the type of training data does not have any significant impact on the machine learning algorithms.

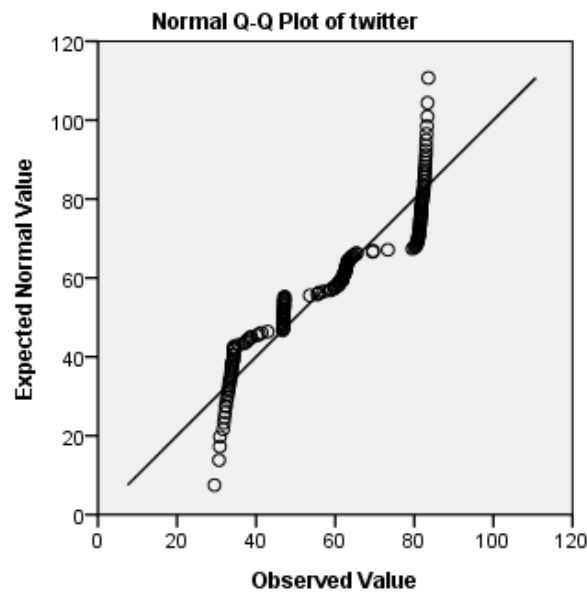
<sup>53</sup> Negative correlation can be defined as the relationship between two variables in which one variable increases as the other decreases and vice versa.

<sup>54</sup> Positive correlation can be defined as the relationship between two variables in which both variables increase and decrease together.

### 4.5.3 Correlation analysis of Twitter and SMS datasets

According to Figures 4.4 and 4.5, two statistical methods could be used for finding any correlation coefficient. Those methods are from Pearson and Spearman. Pearson's correlation coefficient is used when the data has a normal distribution, whereas the opposite is the case for Spearman's correlation coefficient.

Therefore, for selecting the method of correlation coefficient, the distribution of data was tested using the Q-Q plot. The Q-Q plot is a graphic technique for determining the distribution assumption for data. The output from the Q-Q plot is shown in Figures 4.5 and 4.6; we did not achieve a straight line but obtained a wiggle of dots around the line; thereby demonstrating that we may not have normal distribution. This is supported by Field (2013b) who stated that, if the Q-Q plot looks like a straight line with a wiggled snake wrapped around it, then there is some deviation from the normal distribution.



**Figure 4-5:** Q-Q plots of data from Twitter dataset

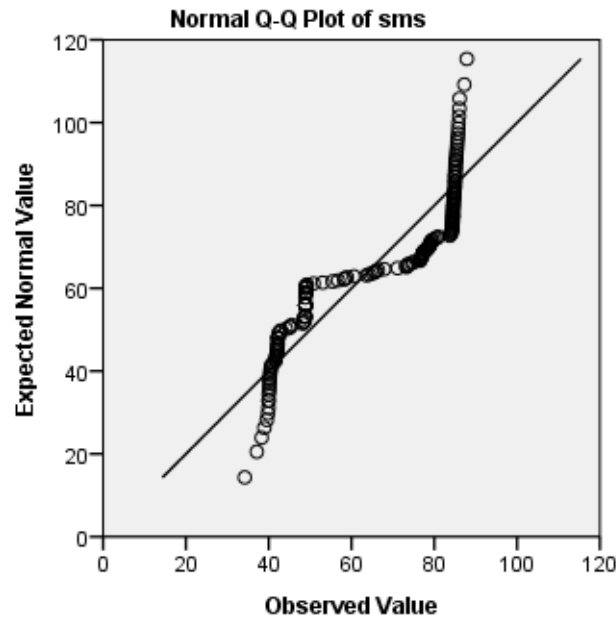


Figure 4-6: Q-Q plots of data from SMS dataset

Due to our data being non-normally distributed, Spearman's correlation coefficient was chosen. The results are presented in Table 4.13.

Correlations				
			twitter	sms
Spearman's rho	twitter	Correlation Coefficient	1.000	.965**
		Sig. (2-tailed)	.	.000
		N	189	189
	sms	Correlation Coefficient	.965**	1.000
		Sig. (2-tailed)	.000	.
		N	189	189

\*\* . Correlation is significant at the 0.01 level (2-tailed).

Table 4-13: Correlation analysis of Twitter and SMS datasets

The results from Spearman's correlation coefficient reveal the value of 0.965 which is significant ( $p < 0.01$  for a 2-tailed test), based on 189 complete observations. Thus, this confirms that there is a good positive correlation coefficient between the Twitter and SMS data.



#### 4.6 Conclusion

This chapter has shown that the combination of all features in data pre-processing achieved a higher degree of accuracy than a subset of these features. This chapter also presented an experimental design to investigate the factors in sentiment analysis datasets. The focuses of this study were the investigation and identification of factors that affect sentiment analysis performance. These factors comprise types of training data and machine learning algorithms. The types of training data consisted of training data obtained from the task (Wilson *et al.*, 2013), sentiment lexicons, lexicons from sentiment resources and a combination of these. The machine learning algorithms considered were Naïve Bayes (Tan *et al.*, 2009), the Support Vector Machine (SVM) (Kecman, 2005) and Maximum Entropy Modelling (MaxEnt) (Harte, 2011).

In the experiment, Twitter contexts were used for both the training and test datasets. The results revealed that by using only the training dataset it was possible to achieve an F-score accuracy of 81.60%, 82.62% and 59.93% using Naïve Bayes (Tan *et al.*, 2009), Support Vector Machine (SVM) (Kecman, 2005) and Maximum Entropy Modelling (MaxEnt) (Harte, 2011), respectively. By using only the lexicon as training data, the results showed that by using the Support Vector Machine (SVM) (Kecman, 2005), AFINN's Lexicon (Nielsen, 2011a) achieved a higher level of accuracy than the others with an F-score of 69.56%. However, the best accuracy performance was achieved by the combination of training data and all sentiment lexicons with an F-score of 83.55% using the Support Vector Machine (SVM) (Kecman, 2005). The reason for this might be that the training dataset contains information related to the testing dataset, while the sentiment lexicons did not contain information related to the task but helped to increase some information missing from the training dataset.

In the evaluation process, the context of SMS was used for the testing dataset, while the training dataset remained the same. The purpose was to observe how well our system could be generalised to other types of testing data. The results revealed that we achieved this well on the SMS dataset; the baseline results obtained an F-score accuracy of 85.49%, 85.05%, 50.80% using Naïve Bayes (Tan *et al.*, 2009), Support Vector Machine (SVM) (Kecman, 2005) and

Maximum Entropy Modelling (MaxEnt) (Harte, 2011), respectively. Moreover, using only a lexicon as training data, the results from AFINN's Lexicon (Nielsen, 2011a) using Naïve Bayes (Tan *et al.*, 2009) achieved a higher degree of accuracy than the other lexicons with an F-score of 74.96%. Nevertheless, the best accuracy performance was achieved by a combination of training data and MPQA's lexicon with an F-score of 87.85% using Naïve Bayes (Tan *et al.*, 2009).

Sentiment resources were used in addition to machine learning algorithms. These were SentiStrength (SS) (Thelwall *et al.*, 2010b) and SentiWordNet (SWN) (Baccianella *et al.*, 2010a). The results from both testing dataset (Twitter and SMS) achieved lower levels of accuracy than using machine learning algorithms by themselves. Moreover, an analysis of variance (ANOVA) was used to analyse the results from both testing datasets (Twitter and SMS). The results showed that different machine learning algorithms affect sentiment analysis performance, and at least one machine learning algorithm achieved lower system performance.

Overall, we can conclude from this experiment that our system performs well and can be generalised to testing data derived from both Twitter and SMS. Training data is essential to the sentiment analysis task as there is a chance that the combination of training data and sentiment lexicons can help to improve the system accuracy performance. Therefore, sentiment lexicons may be essential to sentiment analysis and should be combined with training data. However, in the absence of training datasets, sentiment lexicons can be used instead. We found that AFINN's lexicons achieved better levels of accuracy than the other single sentiment lexicons in both testing datasets (Twitter and SMS). Furthermore, from the results of ANOVA, we can conclude that SVM (Kecman, 2005) is the most effective single machine algorithm, whilst Maximum Entropy Modelling (MaxEnt) (Harte, 2011) is the least effective. Also, there is a strong positive correlation coefficient between Twitter and SMS. The next chapter describes the experiment on the detection of sentiment using ensemble learning techniques.

## Chapter 5 : Novel Ensemble Experiments for Sentiment Analysis

This chapter outlines the details of the investigation and presents an analysis of the theoretical principles by re-contextualising existing algorithms; namely the Arbiter Tree (Chan and Stolfo, 1993) and Combiner Tree (Chan and Stolfo, 1997). Both algorithms fall within the area of ensemble learning and the purpose is to use a combination of multiple machine learning algorithms in order to predict the final results. The aims and purposes of this experiment are to discover whether the Arbiter Tree (Chan and Stolfo, 1993) and Combiner Tree (Chan and Stolfo, 1997) can be used in sentiment analysis and to debate whether an ensemble learning algorithm works better than a single machine learning algorithm. These algorithms have been chosen because, as learned from the literature review, no studies have used the Arbiter Tree (Chan and Stolfo, 1993) and Combiner Tree (Chan and Stolfo, 1997) in sentiment analysis.

Two single machine learning algorithms from the previous experiment (Chapter 4) were selected for use in this experiment: the Support Vector Machine (SVM) (Kecman, 2005) (Section 2.2) and the Naïve Bayes (Tan *et al.*, 2009) (Section 2.2). These were chosen because both of them achieved good accuracy, as demonstrated in the previous chapter. The remainder of this chapter is structured as follows. An overview of ensemble learning is described in Section 5.1. Sections 5.2 and 5.3 describe the overview and implementation of the Arbiter Tree (Chan and Stolfo, 1993) and Combiner Tree (Chan and Stolfo, 1997). The results and discussion are presented in Sections 5.4 and 5.5, respectively.

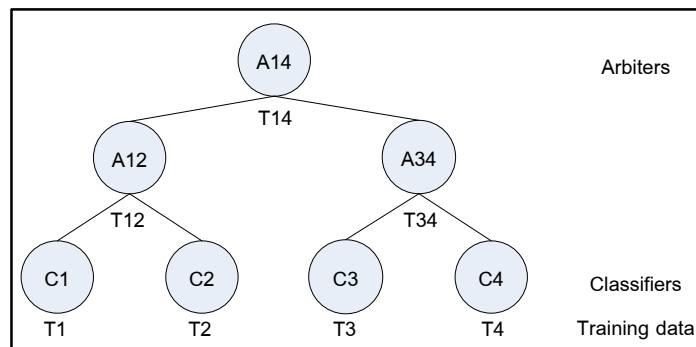
### 5.1 Ensemble Learning Algorithms; Multiple Classifiers

Ensemble learning is an approach to machine learning that uses multiple classifiers to train data and make a final prediction, and often achieves higher accuracy than using a single classifier (Rokach, 2010). However, there is no guarantee that ensemble learning algorithms (Rokach, 2010; Tang *et al.*, 2010) will outperform a single, trained, machine learning algorithm. Ensemble learning methods can be divided into two types: a common method (Section 2.3.1) and a

combining method. There are two methodologies within a combining method: simple combining (Section 2.3.2) and meta-combining methods (Section 2.3.3). Two methodologies from meta-combining methods are relevant. They are the Arbiter Tree (Chan and Stolfo, 1993) and the Combiner Tree (Chan and Stolfo, 1997). As far as can be determined currently, no one has used either the Arbiter Tree (Chan and Stolfo, 1993) or the Combiner Tree (Chan and Stolfo, 1997) in the area of sentiment analysis. Their details are described in the following sections.

## 5.2 Arbiter Tree

The Arbiter Tree (Chan and Stolfo, 1993) method is a meta-combining method of an ensemble learning algorithm which uses training data classified from base classifiers. In the theory process of producing the training data for Arbiter Tree, Chan and Stolfo (1993) mentioned using four training data ( $T$ ) subsets and four classifiers ( $C$ ). After that, the results  $T_1$  and  $T_2$  are combined and selection rules are used to generate a training set for arbiter  $A_{12}$  with the same learning algorithm used in the initial classifiers. This process is similar to arbiter  $A_{34}$ , which uses the training data which results from combining  $T_3$  and  $T_4$  and then produces the first level of arbiter. After obtaining the results from  $T_{12}$  and  $T_{34}$ , these will be combined with  $T_{14}$  to form training data for the root arbiter  $A_{14}$  and the Arbiter Tree is completed, as illustrated in Figure 5.1.



**Figure 5-1:** Theory flowchart to produce a training dataset for Arbiter Tree (Chan and Stolfo, 1993)

There are two strategies that are concerned with the Arbiter Tree (Chan and Stolfo, 1993) method: selection rules and arbitration rules. Selection rules compare the prediction of base classifiers in order to choose the training dataset

for the arbiter. A description of the selection rules is based on two leaf classifiers ( $AT_1$  and  $AT_2$ ) rooted at two sibling arbiters and a training data example,  $E$ ; the strategy generates a set of arbiter training examples,  $T$ .  $AT_i(x)$  denotes the prediction of the training example  $x$  by arbiter sub tree  $AT_i$ .  $class(x)$  denotes the given classification of example  $x$ . There are three versions of the selection rules, described as follows:

The first scheme returns instances with predictions that disagree between base classifiers:  $T = T_d = \{x \in E \mid AT_1(x) \neq AT_2(x)\}$ . (This scheme is denoted as meta-different<sup>55</sup>.)

The second scheme returns instances with predictions that disagree,  $T_d$ , in the first instance but also predictions that agree but are incorrect:  $T = T_d \cup T_i$ , where  $T_i = \{x \in E \mid (AT_1(x) = AT_2(x)) \wedge (class(x) \neq AT_1(x))\}$ . (For further reference, this scheme is denoted as meta-different-incorrect<sup>56</sup>.)

The final scheme returns a set of three training sets:  $T_d$  and  $T_i$ , as above and  $T_c$  with examples that have the same correct predictions:  $T = \{T_d, T_i, T_c\}$  where  $T_c = \{x \in E \mid (AT_1(x) = AT_2(x)) \wedge (class(x) = AT_1(x))\}$ . Note that,  $T_d, T_i$  and  $T_c$  generate  $A_d, A_i$  and  $A_c$ , respectively. (This scheme is denoted as meta-different-incorrect-correct<sup>57</sup>.)

The sample of training data that are generated by these selection rules are illustrated in Figure 5.2.

<sup>55</sup> The following terminology from is CHAN, P. K. 1996. *An extensible meta-learning approach for scalable and accurate inductive learning*. Columbia University

<sup>56</sup> Ibid.

<sup>57</sup> Ibid.

Class	Attribute Vector	Example	Base classifiers' predictions	
$class(x)$	$attrvec(x)$	$x$	$C_1(x)$	$C_2(x)$
a	$attrvec_1$	$x_1$	a	a
b	$attrvec_2$	$x_2$	a	b
c	$attrvec_3$	$x_3$	b	b
b	$attrvec_4$	$x_4$	b	b

Training set from The meta-different arbiter scheme		
Instance	Class	Attribute Vector
1	b	$attrvec_2$

Training set from The meta-different-incorrect arbiter scheme		
Instance	Class	Attribute Vector
1	b	$attrvec_2$
2	c	$attrvec_3$

Training set from The meta-different-incorrect-correct arbiter scheme			
Set	Instance	Class	Attribute Vector
Different ( $T_d$ )	1	b	$attrvec_2$
Incorrect ( $T_i$ )	1	c	$attrvec_3$
Correct ( $T_c$ )	1	a	$attrvec_1$
	2	b	$attrvec_4$

**Figure 5-2:** Sample of training data generated by selection rules (Chan, 1996)

The final prediction is decided based on the base classifiers, arbiter and arbitration rules, as presented in Figure 5.3, using arbitration rules with the aim of learning from incorrect classification (Chan and Stolfo, 1993). There are two versions of arbitration rules, which are described as follows:

The first version of the arbiter rule (ABT1) returns the majority of predictions from base classifiers,  $p_1$  and  $p_2$  and the arbiter,  $A_{(instance)}$ , with preference given to the arbiter's choice;

if  $p_1 \neq p_2$  return  $A_{(instance)}$

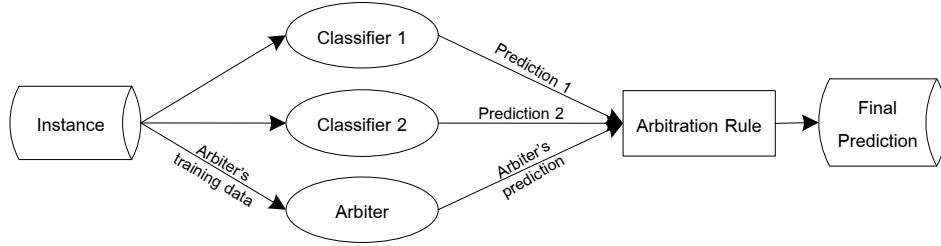
else return  $P_1$ .

The second version of the arbiter rule (ABT2) is similar to the first version, but uses the subset of the arbiter's results instead of  $A_{(instance)}$ ;

```

if  $p_1 \neq p_2$  return  $A_{d(instance)}$ 
else if  $p_1 = A_{c(instance)}$  return  $A_{c(instance)}$ 
else return  $A_{i(instance)}$ 
where  $A = \{A_d, A_i, A_c\}$ 

```



**Figure 5-3:** Theory flowchart of how the final prediction is made in Arbiter Tree (Chan, 1996)

### 5.2.1 Implementation

In our experiment, the Arbiter Tree (Chan and Stolfo, 1993) method was implemented and analysed using SVM (Kecman, 2005) and Naïve Bayes (Tan *et al.*, 2009) as base classifiers. In exhausted nature of the Randomised Complete Block Design (RCBD) (Section 4.2.1) entails multiple experiments repeated since each treatment required approximated one day experimental effect the number of treatments that could be explored is limited. The number of treatments were selecting using a compare mean<sup>58</sup> in SPSS (IBM, 2010). The top five treatments which achieved a better score in the compare mean were chosen, as presented in Table 5.1 (see Appendix VIII for the full table). Besides that, treatment No. 1, which uses only training datasets will be tested and used as the baseline result (benchmark).

<sup>58</sup> The compare mean is a basic statistical method used for computing the average score (mean) of two independent samples.

Comparison of mean			
F-score			
Treatments		Mean <sup>b</sup> (%)	N <sup>c</sup>
No. <sup>a</sup>	Name		
38	TR + MPQA + AFINN	82.47	2
36	TR + HL +AFINN	82.53	2
45	TR + SS + HL + MPQA + AFINN	82.59	2
35	TR + HL + MPQA + AFINN	82.75	2
39	TR + AFINN	82.96	2
Total		68.3554	126

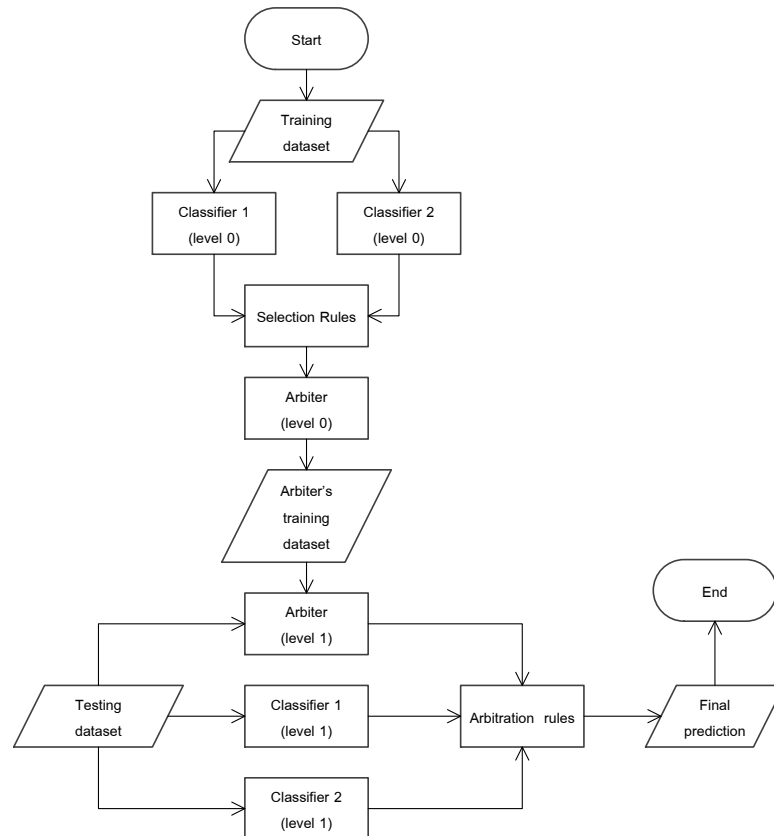
a. No. refers to the number of the treatment which starts from 1 to 63.

b. Mean refers to the means of variables. In this case, they are the results of F-score (%).

c. N refers to the number of treatment used with machine learning algorithms. They showed 2 because the results of them from Support Vector Machine (Kecman, 2005) and Naïve Bayes (Tan *et al.*, 2009).

**Table 5-1:** Comparison of mean of the top five treatments

The overall method pipeline of the Arbiter Tree (Chan and Stolfo, 1993), built in the TJP system, is illustrated in Figure 5.4.



**Figure 5-4:** The overall of Arbiter Tree pipeline in the TJP system



To build the training dataset, two machine learning algorithms were used as base classifiers and the training data was not divided into subsets, as stated in the theory. From Figure 5.4, firstly, the base training dataset was trained with the base classifiers (level 0), by representing training and testing data. Base classifiers (level 0) are SVM (Kecman, 2005) and Naïve Bayes classifiers (Tan *et al.*, 2009). After that, the results were combined and sorted using the selection rules. The selection rules from Chan and Stolfo (1993) have not been used directly but they are expanded and trimmed into four schemes (see the pseudocode in algorithm 1).

The first scheme,  $T_d$ , returned instances with predictions that disagreed between base classifiers.

The second scheme,  $T_i$ , returned instances with predictions that agreed between base classifiers but were incorrect.

The third scheme,  $T_c$ , returned instances with predictions that agreed between base classifiers and were correct.

The final scheme,  $T$ , returned instances that combined the set of  $T = \{T_d, T_i, T_c\}$ . Note that,  $T_d, T_i$  and  $T_c$  generate  $A_d, A_i$  and  $A_c$ , respectively.

The sample of training data that was generated by these selection rules is illustrated in Figure 5.5.

Class	Attribute Vector	Example	Base classifiers' predictions	
$class(x)$	$attrvec(x)$	$x$	$C_1(x)$	$C_2(x)$
a	$attrvec_1$	$x_1$	a	a
b	$attrvec_2$	$x_2$	a	b
c	$attrvec_3$	$x_3$	b	b
b	$attrvec_4$	$x_4$	b	b

Training set from $T_d$		
Instance	Class	Attribute Vector
1	b	$attrvec_2$

Training set from $T_i$		
Instance	Class	Attribute Vector
1	c	$attrvec_3$

Training set from $T_c$		
Instance	Class	Attribute Vector
1	a	$attrvec_1$
2	b	$attrvec_4$

Training set from $T$		
Instance	Class	Attribute Vector
1	a	$attrvec_1$
2	b	$attrvec_2$
3	c	$attrvec_3$
4	b	$attrvec_4$

**Figure 5-5:** Sample of training data generated by selection rules for using in TJP system

Chan and Stolfo (1993) did not discuss clearly how to assign and use the selection rules; therefore, all selection rules were decided and trained with the arbiter (level 0) for creating the final training data for the arbiter (level 1). SVM (Kecman, 2005) was chosen for use as an arbiter for both levels due to the fact that SVM (Kecman, 2005) achieved better accuracy than the Naïve Bayes classifiers (Tan *et al.*, 2009) in the previous experiment (Chapter 4). Next, for processing the final prediction, the classifiers (level 1) were trained by using base training datasets and a testing dataset. On the other hand, the arbiter (level 1) was trained using the arbiter's training dataset but the same testing data will be used. After that, the results were sorted using arbitration rules to deliver the final prediction (see the pseudocode in algorithm 2 and 3). The system diagram of the Arbiter Tree (Chan and Stolfo, 1993) is illustrated in Figure 5.6. The results are analysed and discussed in Sections 4.4 and 4.5, respectively.



---

**Algorithm 2:** Processing final decision from Arbiter Tree (arbiter rule version 1)

---

Data: results from classifier 1 (SVM)  $S$ , classifier 2 (Naïve Bayes)  $N$  and arbiter (SVM)  $A$

Result: final prediction

```
1 # arbiter rule version 1 (ABT 1)
2 foreach data  $I$  that predict polarity as  $s_j \in S$ ,  $n_j \in N$  and  $a_j \in A$ 
3   if  $s_j \neq n_j$ 
4     return  $a_j$ 
5   else
6     return  $s_j$ 
7   end
8 end
```

---

---

**Algorithm 3:** Processing final decision from Arbiter Tree (arbiter rule version 2)

---

Data: results from classifier 1 (SVM)  $S$ , classifier 2 (Naïve Bayes)  $N$  and arbiter (SVM)  $A$

Result: final prediction

```
1 # arbiter rule version 2 (ABT 2)
2 foreach data  $J$  that predict polarity as  $s_j \in S$ ,  $n_j \in N$  and  $a_d, a_i, a_c \in A$ 
3   if  $s_j \neq n_j$ 
4     return  $a_{jd}$ 
5   else if  $s_j = a_{jc}$ 
6     return  $a_{jc}$ 
7   else
8     return  $a_{ji}$ 
9   end
10 end
```

---

### 5.3 Combiner Tree

The Combiner Tree (Chan and Stolfo, 1997) is an algorithm similar to the Arbiter Tree (Chan and Stolfo, 1993); however, the Combiner Tree (Chan and Stolfo, 1997) is trained directly by using the training output from base classifiers that passed composition rules. There are two schemes in the composition rules (Chan and Stolfo, 1997). The description of the composition rules is based on the prediction,  $C_1(x), C_2(x), \dots, C_k(x)$ ; each example  $x$  in the validation set of examples,  $E$ , is generated by the  $k$  base classifiers. These predicted classifications are used to form a new set of meta-level training data,  $T$ , which is used as input for a learning algorithm that computes a combiner.

The first scheme returns meta-training data with the correct classification and the prediction:  $T = \{(class(x), C_1(x), C_2(x), \dots, C_k(x)) \mid x \in E\}$ . (For further reference, this scheme is denoted as class-combiner<sup>59</sup>.)

The second scheme returns meta-training data as above with the addition of attributer vectors:  $T = \left\{ \left( \begin{matrix} class(x), C_1(x), C_2(x), \dots, C_k(x), \\ attribute\_vector(x) \end{matrix} \right) \mid x \in E \right\}$ . (For further reference, this scheme is denoted as class-attribute-combiner<sup>60</sup>.)

The final prediction is decided based on the base classifiers and combiner, as presented in Figure 5.8. The sample of training data generated by these composition rules is given in Figure 5.7.

---

<sup>59</sup> Ibid.

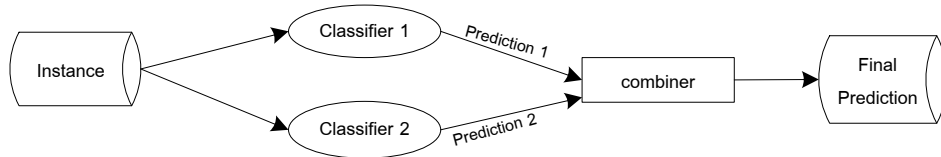
<sup>60</sup> Ibid.

Class	Attribute Vector	Example	Base classifiers' predictions	
$class(x)$	$attrvec(x)$	$x$	$C_1(x)$	$C_2(x)$
a	$attrvec_1$	$x_1$	a	a
b	$attrvec_2$	$x_2$	a	b
c	$attrvec_3$	$x_3$	b	b
b	$attrvec_4$	$x_4$	b	b

Training set from The class-combiner scheme		
Instance	Class	Attribute Vector
1	a	(a, a)
2	b	(a, b)
3	c	(b, b)
4	b	(b, b)

Training set from The class-attribute-combiner scheme		
Instance	Class	Attribute Vector
1	a	(a, a, $attrvec_1$ )
2	b	(a, b, $attrvec_2$ )
3	c	(b, b, $attrvec_3$ )
4	b	(b, b, $attrvec_4$ )

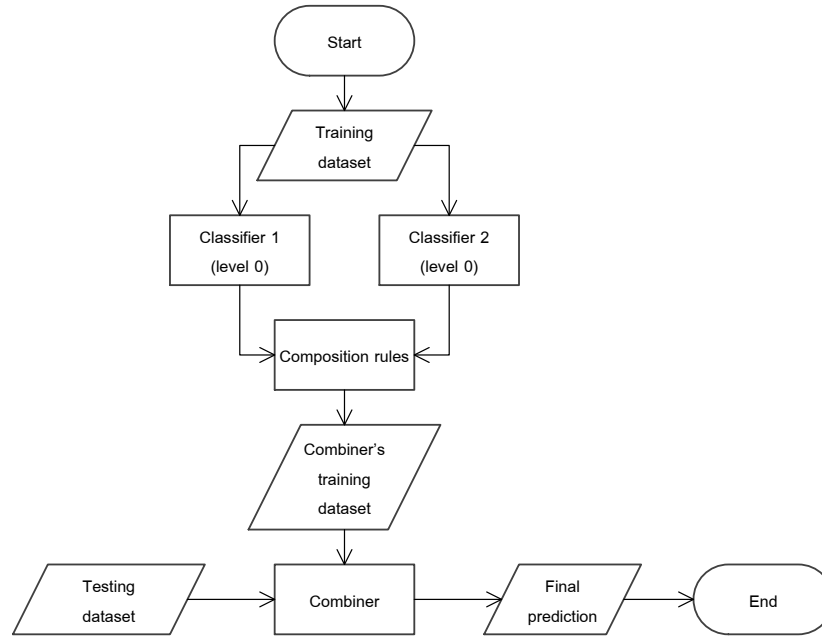
**Figure 5-7:** Sample of training data that generated by composition rules (Chan, 1996)



**Figure 5-8:** Theory flowchart of how the final prediction is made in Combiner Tree (Chan, 1996)

### 5.3.1 Implementation

In our experiment, the Combiner Tree (Chan and Stolfo, 1997) was implemented and analysed using SVM (Kecman, 2005) and Naïve Bayes (Tan *et al.*, 2009) as base classifiers, the same as for the Arbiter Tree (Chan and Stolfo, 1993). In addition, five treatments were used, the same as for the Arbiter Tree (Chan and Stolfo, 1993). The overall method pipeline of the Combiner Tree (Chan and Stolfo, 1997), built in the TJP system, is illustrated in Figure 5.9.



**Figure 5-9:** The overall of Combiner Tree pipeline in the TJP system

For building the training dataset, two machine learning algorithms were used as base classifiers and the training data was not divided into subsets. From Figure 5.9, firstly, the base training dataset was trained with base classifiers (level 0), represented as training and testing data. The base classifiers (level 0) are SVM (Kecman, 2005) and Naïve Bayes classifiers (Tan *et al.*, 2009). After that, the results were combined and sorted using the composition rules for generating the combiner's training datasets (see the pseudocode in algorithm 4). Next, the combiner's training datasets were trained with the combiner to produce the final prediction. SVM (Kecman, 2005) was chosen to use as the combiner. The reason for choosing SVM (Kecman, 2005) is the same as discussed in Section 5.2.1. The system diagram of the Combiner Tree (Chan and Stolfo, 1997) is illustrated in Figure 5.10. The results are analysed and discussed in Sections 4.4 and 4.5, respectively.

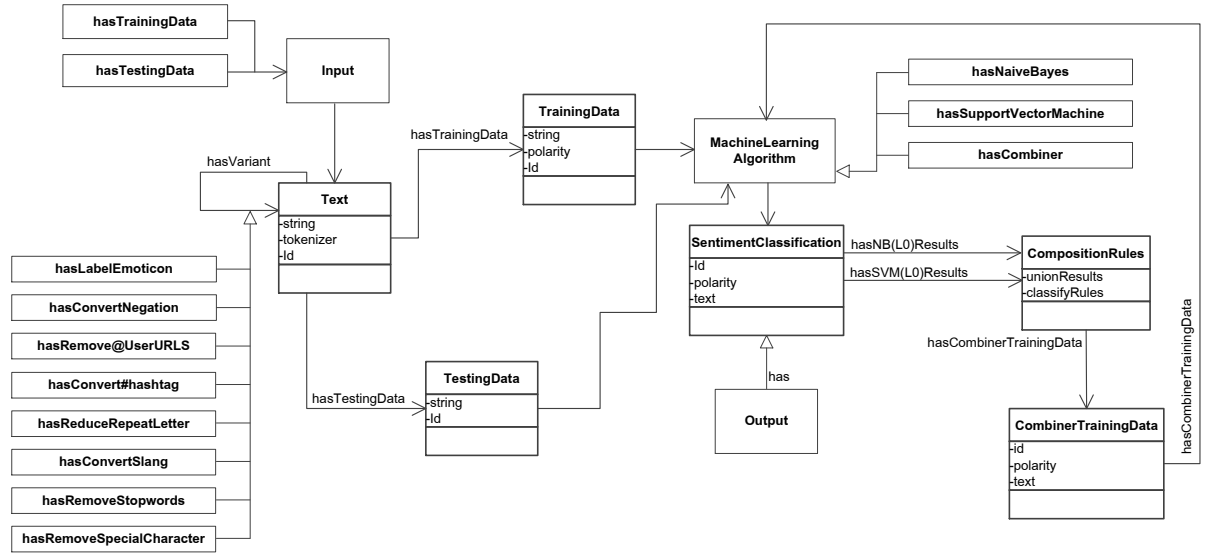


Figure 5-10: Diagram of Combiner Tree in TJP system

---

**Algorithm 4:** Creating training data for combiner

---

Data: results from classifier 1 (SVM) S and classifier 2 (Naïve Bayes) N and correct polarity P

Result: training dataset for combiner

```

1  foreach data I that predict polarity as  $s_i \in S$ ,  $n_i \in N$  and has correct polarity as  $p_i \in P$ 
2    for data I
3      # composition rule Class-combiner
4      # composition rule version 1 (CBT 1)
5      return  $p_i, s_i$  and  $n_i$ 
6      # composition rule Class-attribute-combiner
7      # composition rule version 2 (CBT 2)
8      return  $p_i, s_i, n_i$  and I
9    end
10 end

```

---



#### 5.4 Analysis of Results

The F-score evaluation metric (Section AA) was used as in the previous experiment (Chapter 4). A comparison between the Arbiter Tree (Chan and Stolfo, 1993) and the Combiner Tree (Chan and Stolfo, 1997) classifiers is presented in Table 5.2. After comparing the results with the baseline, the performance accuracy appears to have improved when using the second rule arbiter of the Arbiter Tree (Chan and Stolfo, 1993) in both datasets. Meanwhile, the best results for the Combiner Tree (Chan and Stolfo, 1997) were from the baseline.

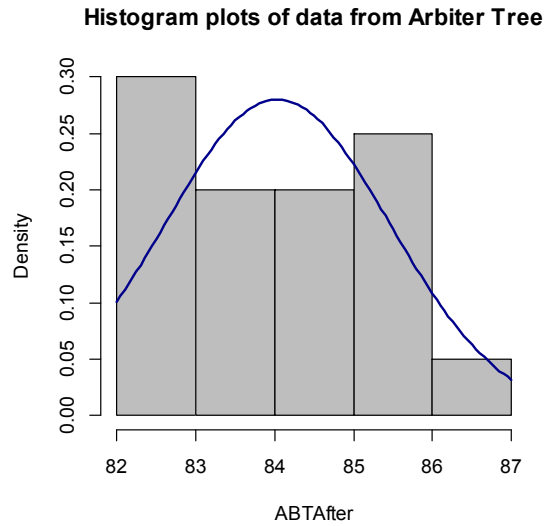
Treatments	Twitter				SMS			
	ABT	ABT	CBT	CBT	ABT	ABT	CBT	CBT
	(1)	(2)	(1)	(2)	(1)	(2)	(1)	(2)
1 (baseline)	81.55	81.24	62.38	63.43	84.97	85.31	41.41	70.33
35	82.31	<b>83.58</b>	30.25	30.90	84.87	85.56	34.59	34.65
36	82.08	83.56	30.25	31.12	84.69	85.61	34.59	34.76
38	82.00	83.34	30.25	31.82	85.15	85.23	34.59	34.63
39	82.67	82.92	30.25	41.00	84.86	<b>86.84</b>	34.59	41.28
45	82.09	83.28	30.25	30.23	84.82	85.24	34.59	34.68

**Table 5-2:** The results of Arbiter and Combiner Trees

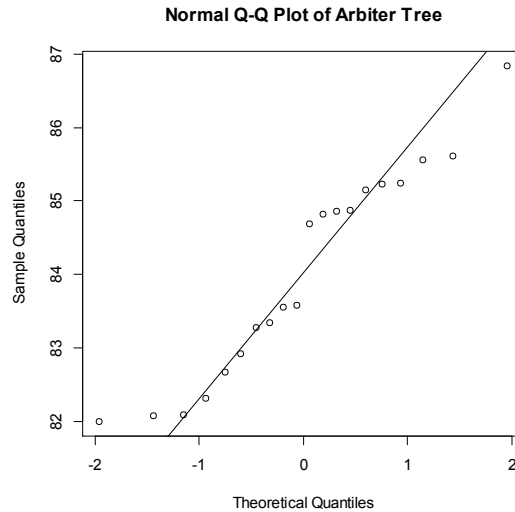
This improvement in the results however might be attributed solely to chance. The traditional approach to addressing this issue is to perform a test for statistical significance. There are methods in statistics that can be used. In order to choose a suitable method, the results from the Arbiter Tree<sup>61</sup> (Chan and Stolfo, 1993) were tested for distribution using a histogram, as shown in Figure 5.11. The graph illustrates that the distribution might be as normal. Furthermore, a Q-Q plot (Section 4.5.3) is used to check whether the distribution is normal. The output from the Q-Q plot is given in Figure 5.12; we did not achieve a straight line, but there was a little wiggle of dots around the line which could be seen as normal distribution. However, Shapiro-Wilk Normality Test (Shapiro and Wilk, 1965) was used for supporting. The results showed w-value = 0.9319, p-value = 0.168 >> 0.05 so, the data is normally distributed

<sup>61</sup> The results from Combiner Tree have not been used because the results are lower than baseline that achieved from using single machine learning algorithm in Chapter 3.

Besides the distribution, this experiment is composed of one dependent variable, which is the score of Arbiter Tree (Chan and Stolfo, 1993), and two independent variables, which are the before and after scores of the Arbiter Tree (Chan and Stolfo, 1993). We assume that the before scores are those from the baseline. Conversely, the after scores are those that Arbiter Tree (Chan and Stolfo, 1993) tested using five treatments (Section 5.2.1).



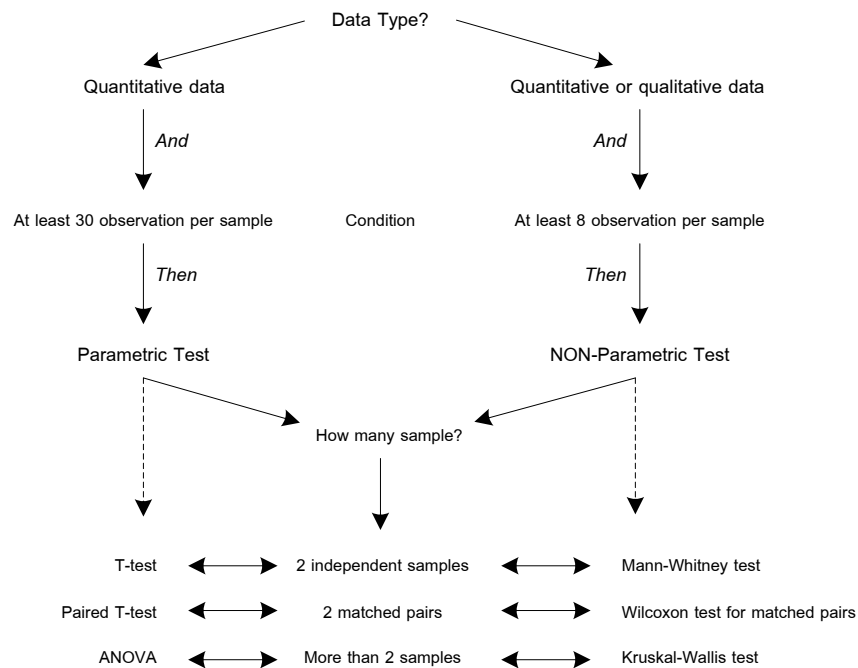
**Figure 5-11:** Histogram plots of data from Arbiter Tree from R



**Figure 5-12:** Q-Q plots of data from Arbiter Tree

Although, the data have a normal distribution but our sample size is less than 30, the statistical methods for a non-parametric test will be chosen. A non-parametric test is used when assumptions cannot be made regarding the normality

of the population distribution or if the sample size is too small. Conversely, if the sample size is greater than 30, a parametric test will be used (Grant and Tomal, 2013). From the Baran and Warry (2008) flowchart (see Figure 5.14), we found that, in order to examine the pair of scores (before and after) of a non-parametric test, the Wilcoxon signed-ranks test (Wilcoxon, 1945) (non-parametric test) is the most appropriate. Thus, a Wilcoxon signed-ranks test (Wilcoxon, 1945) was used in this experiment. This particular test (Wilcoxon, 1945) is an appropriate statistical method for comparing two populations of ordinals when the data is generated from independent samples. Moreover, the Wilcoxon signed-ranks test (Wilcoxon, 1945) can be used when you do not want to assume that the difference between results is normally distributed (Field, 2013c).



**Figure 5-13:** Flowchart for choosing the appropriate statistical test (Baran and Warry, 2008)

In our case, the Wilcoxon signed-ranks test (Wilcoxon, 1945) in SPSS (IBM, 2010) was used for testing at an 0.05 significance level, where indicated. Table 5.3 shows the ranking process for the data used in the Wilcoxon signed-ranks test (Wilcoxon, 1945). To establish the rank, we first calculated the differences between the scores (before and after). The sign of the differences was

noted as positive (+) or negative (-). Next, the differences were assigned potential ranks from the smallest to the largest by ignoring whether they were positive or negative. The rank is referred to as the potential rank because sometimes the same scores occur more than once in the data (e.g. in this data a score of 0.18 occurs twice). These values need to be assigned the same rank by using the average. Therefore, the two scores of 0.18 have potential ranks of 6 and 7 and the average of these values is 6.5. Therefore, 6.5 is used as an actual rank for both occurrences of the score. Finally, each actual rank was assigned to its group to calculate the significance of the test statistic, regardless of whether they were positive or negative (Field, 2013c). The output is presented in Table 5.4.

Before	After	Difference	Sign	Potential Rank	Actual Rank	Positive Ranks	Negative Ranks
<b>Arbiter Tree</b>							
81.55	82.31	0.76	+	13	13	13	
81.55	82.08	0.53	+	11	11	11	
81.55	82.00	0.45	+	10	10	10	
81.55	82.67	1.12	+	14	14	14	
81.55	82.09	0.54	+	12	12	12	
81.24	83.58	2.34	+	20	20	20	
81.24	83.56	2.32	+	19	19	19	
81.24	83.34	2.10	+	18	18	18	
81.24	82.92	1.68	+	16	16	16	
81.24	83.28	2.04	+	17	17	17	
84.97	84.87	-0.10	-	3	3		3
84.97	84.69	-0.18	-	7	6.5		6.5
84.97	85.15	0.18	+	6	6.5	6.5	
84.97	84.86	-0.11	-	4	4		4
84.97	84.82	-0.15	-	5	5		5
85.31	85.56	0.25	+	8	8	8	
85.31	85.61	0.30	+	9	9	9	
85.31	85.23	-0.08	-	2	2		2
85.31	86.84	1.53	+	15	15	15	
85.31	85.24	-0.07	-	1	1		1

**Table 5-3:** Ranking data in the Wilcoxon signed-ranks test

Ranks		N	Mean Rank	Sum of Ranks
After - Before	Negative Ranks	6 <sup>a</sup>	3.83	23.00
	Positive Ranks	14 <sup>b</sup>	13.36	187.00
	Ties	0 <sup>c</sup>		
	Total	20		
a. After < Before				
b. After > Before				
c. After = Before				

Test Statistics <sup>a</sup>	
	After - Before
Z	-3.061 <sup>b</sup>
Asymp. Sig. (2-tailed)	.002
a. Wilcoxon Signed Ranks Test	
b. Based on negative ranks.	

**Table 5-4:** Output of Arbiter Tree from Wilcoxon signed-ranks test

The first table provides information about the ranked scores. The table shows that 14 out of 20 scores had improved since the before scores. There are no ties in the ranking. Ties refer to a score that is the same both before and after (Field, 2013c). The table also shows the average number and the sum of negative and positive ranks. Below, the table shows the relationship between the positive and negative ranks.

The second table reveals that the test statistics are based on the negative ranking, the z-score is -3.061 and this value is significant at  $p = 0.002$ . The value of the z-score is greater than 1.96 so we can conclude that the improvement over the baseline is statistically significant.

The z-score is a statistical measurement of standard deviation. The associate z-score is 1.96 as this is the approximate value of the 97.5 percentile point of the normal distribution used in probability and statistics. 95% of the area, under a normal curve, lies within roughly 1.96 standard deviations of the mean (due to the central limit theorem); this number is used in the construction of approximately 95% confidence intervals. Its ubiquity is due to the arbitrary but common convention of using confidence intervals with 95% coverage, rather than other levels of coverage (such as 90% or 99%) (Healey, 2012). The central limit theorem is a theorem in statistics based on sampling the distribution of means.

The theorem states that if the sample size is large (over 30), the sampling distribution of the sample mean is approximately normal (Field, 2013a). Levine and Stephan (2009) and (Black, 2011) mentioned that a sample size of at least 30 is large enough to appear as an approximately normal distribution.

## 5.5 Discussion

The results of Arbiter tree (Chan and Stolfo, 1993) were compared with the results of single machine learning in Chapter 3, as show in Table 5.5. The results showed that Arbiter tree (Chan and Stolfo, 1993) outperform<sup>62</sup> the single learning algorithm, SVM (Kecman, 2005).

Treatments	Twitter				SMS			
	ABT (1)	ABT (2)	NB	SVM	ABT (1)	ABT (2)	NB	SVM
1 (baseline)	81.55	81.24	81.06	82.62	84.97	85.31	85.49	86.05
35	82.31	<b>83.58</b>	81.73	83.20	84.87	85.56	84.84	85.96
36	82.08	83.56	81.74	83.32	84.69	85.61	84.98	85.36
38	82.00	83.34	81.84	83.33	85.15	85.23	84.98	<b>85.78</b>
39	82.67	82.92	81.94	<b>83.55</b>	84.86	<b>86.84</b>	85.03	84.95
45	82.09	83.28	82.91	83.00	84.82	85.24	87.25	85.05

**Table 5-5:** The results of Stacking

The Wilcoxon signed-ranks test (Wilcoxon, 1945) (Section 5.4) was used to test whether this improvement was significant or not. The statistic (Table 5.6) reveals that the test is based on the negative ranking, z-score = -0.105 and p-value = 0.917 for the Twitter and z-score = -0.943 and p-value = 0.345 for the SMS. The value of the z-score is lower than 1.96 (Section 5.4) so we can conclude that the improvement of Arbiter Tree (Chan and Stolfo, 1993) over the single machine learning algorithm was not statistically significant.

<sup>62</sup> This comparison is based on treatments 1, 35, 36, 38, 39 and 45. Due to the time constraint (Section 5.2.1), all treatments cannot be compared.

		Ranks		
		N	Mean Rank	Sum of Ranks
Twitter_After –	Negative Ranks	2 <sup>a</sup>	5.50	11.00
Twitter_Before	Positive Ranks	4 <sup>b</sup>	2.50	10.00
	Ties	0 <sup>c</sup>		
	Total	6		
SMS_After -	Negative Ranks	4 <sup>d</sup>	3.75	15.00
SMS_Before	Positive Ranks	2 <sup>e</sup>	3.00	6.00
	Ties	0 <sup>f</sup>		
	Total	6		

a. Twitter\_After < Twitter\_Before  
 b. Twitter\_After > Twitter\_Before  
 c. Twitter\_After = Twitter\_Before  
 d. SMS\_After < SMS\_Before  
 e. SMS\_After > SMS\_Before  
 f. SMS\_After = SMS\_Before

Test Statistics <sup>a</sup>		
	Twitter_After - Twitter_Before	SMS_After - SMS_Before
Z	-.105 <sup>b</sup>	-.943 <sup>b</sup>
Asymp. Sig. (2-tailed)	.917	.345

a. Wilcoxon Signed Ranks Test  
 b. Based on positive ranks.

**Table 5-6:** Output of the comparison of Arbiter Tree and SVM from Wilcoxon signed-ranks test

Furthermore, the Arbiter Tree (Chan and Stolfo, 1993) was test again Stacking (Wolpert, 1992) (Section 2.3.3) and Majority Voting (Polikar, 2012) (Section 2.3.2) with the intent of comparing the ensemble learning algorithms that are commonly used with the algorithms that have never been used in sentiment analysis. Some researchers that have used Stacking (Wolpert, 1992) or/and Majority Voting (Polikar, 2012) are Wan (2008), Gryc and Moilanen (2014) and Martin-Valdivia *et al.* (2013). Brief outlines of their works are provided in Sections 2.1.6, 2.3.2 and 2.3.3, respectively. The Stacking (Wolpert, 1992) algorithm in WEKA (Hall *et al.*, 2009) (Section 2.2.1 was used to combine SVM (Kecman, 2005) and Naïve Bayes classifiers (Tan *et al.*, 2009). There are two version of Stacking (Wolpert, 1992). In the first version (V01), Naïve Bayes (Tan *et al.*, 2009) was used as the classifier 0 while SVM (Kecman, 2005) was used as the classifier 1, and vice versa for the second version (V02). Overall, the results of the second version were better than the first version, as presented in Table 5.7.

Treatments	Twitter		SMS	
	NB/SVM (V01)	SVM/NB (V02)	NB/SVM (V01)	SVM/NB (V02)
1 (baseline)	62.22	82.38	69.18	85.47
35	26.34	<b>84.14</b>	33.77	87.19
36	35.85	83.51	44.09	87.01
38	32.48	83.52	42.32	<b>87.33</b>
39	62.32	82.77	69.51	86.67
45	35.98	83.63	44.15	87.01

Table 5-7: The results of Stacking

In addition to Stacking (Wolpert, 1992), we implemented the Majority Voting (Polikar, 2012) algorithm in Python by using the same treatments as in Section 4.2.2. There was a problem when using two voters (SVM (Kecman, 2005) and Naïve Bayes (Tan *et al.*, 2009)) when half of the voters are not agreed. This could be solved by using two conditions from Martin-Valdivia *et al.* (2013). The first condition (V01), positive, is used to represent the answers if they are not equal while negative would be used in the second condition (V02). Examples of these conditions are illustrated in Table 5.8. The results of Twitter and SMS (Table 5.9) achieved accuracy at 83.95% and 86.62%, respectively. Both of these were slightly lower than for Stacking (Wolpert, 1992). From the overall results, we found that both results from Stacking (Wolpert, 1992) and Majority Voting (Polikar, 2012) yielded better scores than the baseline results in our tasks (Section 3.8 and 3.9 for the baseline results of Twitter and SMS, respectively).

ID	SVM	NB	1 <sup>st</sup> con (V01)	2 <sup>nd</sup> con (V02)
1	Positive	Positive	Positive	Positive
2	Negative	Negative	Negative	Negative
3	Negative	Positive	<i>Positive</i>	<i>Negative</i>
4	Negative	Negative	Negative	Negative
5	Negative	Positive	<i>Positive</i>	<i>Negative</i>

Table 5-8: Example of the two voters with the conditions



Treatments	Twitter		SMS	
	V01	V02	V01	V02
1 (baseline)	81.68	81.96	83.89	<b>86.62</b>
35	83.82	81.65	85.15	85.60
36	83.75	81.27	85.15	85.74
38	83.62	81.27	84.97	85.86
39	83.21	82.66	85.80	86.33
45	<b>83.95</b>	81.16	85.15	85.12

**Table 5-9:** The results of Majority Voting

Next, the results from Arbiter Tree (Chan and Stolfo, 1993) and Combiner Tree (Chan and Stolfo, 1997) were compared with Stacking (Wolpert, 1992) and Majority Voting (Polikar, 2012). We found that the results for the Combiner Tree (Chan and Stolfo, 1997) were the worst and did not improve accuracy performance. The best results from Arbiter Tree (Chan and Stolfo, 1993) achieved slightly lower accuracy performance using Twitter than Majority Voting (Polikar, 2012) and Stacking (Wolpert, 1992) at F-score 83.58%, 83.95% and 84.14%, respectively. Conversely, the results from Arbiter Tree (Chan and Stolfo, 1993) achieved slightly higher accuracy performance when using SMS than Majority Voting (Polikar, 2012), but lower than Stacking (Wolpert, 1992) at F-score 86.84%, 86.62% and 87.33%, respectively.

## 5.6 Conclusion

This chapter has introduced existing techniques in ensemble learning algorithms ; namely, Arbiter Tree (Chan and Stolfo, 1993) and Combiner Tree (Chan and Stolfo, 1997). The two single machine learning algorithms chosen for use with Arbiter Tree (Chan and Stolfo, 1993) and Combiner Tree (Chan and Stolfo, 1997) were SVM (Kecman, 2005) and Naïve Bayes (Tan *et al.*, 2009). The method of Arbiter Tree (Chan and Stolfo, 1993) and Combiner Tree (Chan and Stolfo, 1997) to determine the sentiment of datasets are explained clearly and in detail. Both Arbiter Tree (Chan and Stolfo, 1993) and Combiner Tree (Chan and Stolfo, 1997) were demonstrated as being appropriate for use in sentiment analysis. The results from Arbiter Tree (Chan and Stolfo, 1993) performed better than the baseline results and Combiner Tree (Chan and Stolfo, 1997). Moreover, the statistical

evaluation suggests that, Arbiter Tree (Chan and Stolfo, 1993) significantly achieve better accuracy than Combiner Tree (Chan and Stolfo, 1997). Their results were compared with single machine learning algorithm. The results showed that Arbiter tree (Chan and Stolfo, 1993) outperform single learning algorithm. In contrast, the improvement is not statistically significantly. Moreover, the results of Arbiter Tree (Chan and Stolfo, 1993) and Combiner Tree (Chan and Stolfo, 1997) were compared with the existing algorithms that are commonly used in sentiment analysis: Stacking (Wolpert, 1992) and Majority Voting (Polikar, 2012). We found that the best results from Arbiter Tree (Chan and Stolfo, 1993) achieved slightly lower accuracy performance using Twitter than Majority Voting (Polikar, 2012) and Stacking (Wolpert, 1992) (F-scores 83.58%, 83.95% and 84.14%, respectively). Conversely, the results from Arbiter Tree (Chan and Stolfo, 1993) achieved slightly higher accuracy performance when using SMS than Majority Voting (Polikar, 2012), but lower than Stacking (Wolpert, 1992) at F-score 86.84%, 86.62% and 87.33%, respectively. However, the results that we observed from Arbiter Tree (Chan and Stolfo, 1993) and Combiner Tree (Chan and Stolfo, 1997) are encouraging when considering whether they are appropriate for use in sentiment analysis. Following this, it would be worth trying Arbiter Tree (Chan and Stolfo, 1993) in sentiment analysis. For example, although Stacking and Majority Voting are commonly used, there is always a chance that Stacking (Wolpert, 1992) might perform better than Majority Voting (Polikar, 2012) or vice versa. This is supported by Martin-Valdivia *et al.* (2013) and Gryc and Moilanen (2014). Martin-Valdivia *et al.* (2013) (Section 2.3.3) presented evidence that the results from Stacking (Wolpert, 1992) achieved slightly higher results than Majority Voting (Polikar, 2012). On the other hand, Gryc and Moilanen (2014) (Section 2.3.2) found that Majority Voting (Polikar, 2012) achieved higher results than Stacking (Wolpert, 1992).

Fernández-Delgado *et al.* (2014) compared the performance accuracy of 179 classifiers by using 121 datasets in multiple domains. Fernández-Delgado *et al.* (2014) showed that the Random Forest (RE) (Section 2.3.1) implemented in R<sup>63</sup> (R Core Team, 2015) achieved the best accuracy at 94.1% Percentage of the

---

<sup>63</sup> R is free programming software for statistical computing and graphics.

Maximum Accuracy (PMA). PMA is a measurement achieved by each classifier, averaged over the whole collection of the datasets (Fernández-Delgado *et al.*, 2014). Whilst, Fernández-Delgado *et al.* (2014) found that the Random Forest (RE) (Section 2.3.1) implemented in R performed better on the average of all their datasets. There is no guarantee that the Random Forest (RE) (Section 2.3.1) implemented in R will perform better when using the same dataset as in this thesis for sentiment analysis. However, it would be useful to investigate this in the future. This was not investigated in this thesis as Fernández-Delgado *et al.* (2014) was not found until after thesis was submitted.

## Chapter 6 : Conclusions and Future Work

The field of sentiment analysis has been described in detail, whilst the different ways in which the area may be approached has also been explored. Brief details of each chapter are summarised in this chapter, followed by outlines of the contribution. Moreover, recommendations for future work in the area of sentiment analysis are also discussed.

### 6.1 Thesis Summary

Firstly, Chapter 1 introduced the idea of sentiment and described the general background of sentiment analysis. Following from this, the motivation behind the decision to investigate sentiment analysis was introduced. After that, the aims and objectives were presented.

Chapter 2 reviewed the status of sentiment analysis. We also reviewed relevant Information Retrieval research literature, with a particular focus on sentiment analysis. It was found that sentiment analysis can be separated into four levels: word-level, phrase-level, sentence-level and document-level. It was also noted that the purpose of sentiment analysis is to identify opinions or attitudes in terms of polarity. It was noted that there are ways that could be used for measuring polarity measurement and labelling classification: the polarity of words, human annotators, emoticons, features-based, range of polarity and sentiment resources and sentiment lexicons. Work in the fields of single machine learning algorithms and ensemble learning algorithms that are used in sentiment analysis were also reviewed. Moreover, natural language packages that contain machine learning algorithms and some real-world techniques and applications that rely on machine learning algorithms were reviewed.

The system design and architecture of our TJP system were presented in Chapter 3. The system is comprised of five groups of elements: datasets, sentiment lexicons, two sentiment resources, three machine learning algorithms and seven features in the data pre-processing process. The datasets were received from SemEval 2013 Task 2A (Wilson *et al.*, 2013). The main dataset consisted of tweets, while SMS data was used to evaluate our system. The polarity used in

these datasets was word polarity (positive and negative). The sentiment lexicons were the Bing Liu Lexicon (HL) (Hu and Liu, 2004), MPQA Subjective Lexicon (MPQA) (Wilson *et al.*, 2005b) and AFINN Lexicon (AFINN) (Nielsen, 2011a). Two sentiment resources were SentiStrength (Thelwall *et al.*, 2010b) and SentiWordNet (Baccianella *et al.*, 2010a). The three machine learning algorithms were Naïve Bayes (Tan *et al.*, 2009), Support Vector Machine (Kecman, 2005) and Maximum Entropy Modelling (Harte, 2011). Seven features in the data pre-processing process were emoticons, negations, Twitter features, repeated letters, slang words, stopwords and special characters. In addition, in the TJP system, there are two sentiment resource approaches. In the first approach, the sentiment resources were used directly with testing datasets. The lexicons of sentiment resources were used with single machine learning algorithms in the second approach. Moreover, they investigated the use of both training and testing dataset with Naïve Bayes (Tan *et al.*, 2009) to find the most suitable features. We found that, the combination of seven features achieved the better accuracy and suitability. Therefore, this combination was used throughout. The evaluation score that was used is the F-measure.

Chapter 4 introduced the factorial experimental design (Montgomery, 2013b) which was used for the TJP system. A randomised complete block design was used for analysis in the factorial experimental design. The main factors which were focused on in this experiment were datasets, sentiment lexicons, sentiment resources and machine learning algorithms. Following from this Chapter 5 presented another experiment where the focus was on using an ensemble learning algorithm which required using the combination of two or more single machine learning algorithms. Therefore, two single machine learning algorithms from the previous experiment were chosen: the Naïve Bayes (Tan *et al.*, 2009) and SVM (Kecman, 2005).

The research questions that were answered in Chapter 4 and 5 are explained and described in the following section.

## 6.2 Research Questions and Answers

In this section the thesis research questions are re-iterated and answered.

**RQ. 1:** ‘How much accuracy in the context of data will be achieved when using SentiStrength (Thelwall et al., 2010b) and SentiWordNet (Baccianella et al., 2010a)? Moreover, will the accuracy be better than the results from word polarity (positive and negative)?’ (as indicated in Section 2.1.3.6)

To answer this question (Chapter 4), two approaches described in Section 3.5 were used with SentiStrength (Thelwall et al., 2010b) and SentiWordNet (Baccianella et al., 2010a). In the first approach, the data was trained directly to the sentiment resources. Both of them achieved an F-score of 78.37% and 72.99% using Twitter and 79.83% and 78.85% using SMS. The second approach used their lexicons as training data. They achieved an F-score of 37.64% and 62.79% using Twitter and 42.71% and 65.83%, using SMS. When comparing the two approaches, the performance accuracy when using both sentiment resources directly achieved better results than when using their lexicons (positive and negative).

**RQ. 2:** ‘Are sentiment lexicons essential in sentiment analysis?’ (as indicated in Section 2.1.3.7)

To answer this question (Chapter 4), three sentiment lexicons that were mostly used in SemEval 2013 Task 2A were chosen. They are AFINN (Nielsen, 2011a), MPQA (Wilson et al., 2005b) and HL (Hu and Liu, 2004). It was found that the AFINN Lexicon (Nielsen, 2011a) achieved better accuracy than the others at F-score 69.56% and 74.96% from Twitter and SMS datasets, respectively.

**RQ. 3:** ‘How much accuracy will be achieved in the context of data if only using training data?’ (as indicated in Section 2.1.3.7)

To answer this question (Chapter 4), the original training data from the SemEval 2013 was used together with three machine learning algorithms which are Naïve Bayes (Tan et al., 2009), Support Vector Machine (SVM) (Kecman, 2005) and Maximum Entropy Modelling (MaxEnt) (Harte, 2011). The results showed that by using only training datasets, the performance accuracy of Twitter testing data achieved an F-score of 81.06%, 82.62% and 59.93% from Naïve Bayes (Tan et al., 2009), Support Vector Machine (SVM) (Kecman, 2005) and

Maximum Entropy Modelling (MaxEnt) (Harte, 2011), respectively. In addition, the performance accuracy of SMS testing data achieved an F-score of 85.49%, 85.05% and 50.80% from Naïve Bayes (Tan et al., 2009), Support Vector Machine (SVM) (Kecman, 2005) and Maximum Entropy Modelling (MaxEnt) (Harte, 2011), respectively.

**RQ. 4:** ‘Will the accuracy improve if using the combination of training data and sentiment lexicon(s)?’ (as indicated in Section 2.1.3.7)

To answer this question (Chapter 4), sentiment lexicons were combined with training dataset. The results showed that after combining training data with sentiment lexicons, the performance accuracy improved in both datasets (Twitter and SMS) at F-scores, 83.55% and 87.85%, respectively.

**RQ. 5:** ‘Which single machine learning is essential in the context of data classifiers between Naïve Bayes (Tan et al., 2009), Support Vector Machine (Kecman, 2005) and Maximum Entropy Modelling (Harte, 2011)?’ (as indicated in Section 2.2.3).

Within Chapter 4, we found that the Support Vector Machine (Kecman, 2005) achieved better accuracy than the others, while, Maximum Entropy Modelling (Harte, 2011) was the worst for both datasets (Twitter and SMS).

**RQ. 6:** ‘If the ensemble learning will be used in the context of, data classifiers, will the accuracy achieved be better than single machine learning algorithm(s)?’ (as indicated in Section 2.3)

To answer this question (Chapter 5), we selected two algorithms from ensemble learning; the Arbiter Tree (Chan and Stolfo, 1993) and Combiner Tree (Chan and Stolfo, 1997). The reason behind the choice is that there is no existing literature to review their ability in the task of sentiment analysis. Therefore, we wished to analyse and investigate them in more detail.

**RQ. 7:** ‘Will the Arbiter Tree (Chan and Stolfo, 1993) and Combiner Tree (Chan and Stolfo, 1997) achieve better performance in the sentiment task within the data context? (as indicated in Section 2.3.3)

From the results of the Arbiter Tree (Chan and Stolfo, 1993) and the Combiner Tree (Chan and Stolfo, 1997) in Chapter 5, we discovered that the Arbiter Tree (Chan and Stolfo, 1993) achieved better accuracy than single

machine learning algorithms but vice versa for the Combiner Tree (Chan and Stolfo, 1997). This result is supported by the statement from Chan (1996) that the Combiner Tree (Chan and Stolfo, 1997) does not perform as well as the Arbiter Tree (Chan and Stolfo, 1993) because of the lack of information in the training data that is trained using the Combiner Tree (Chan and Stolfo, 1997). In addition, Rokach (2010) and Tang et al. (2010) also stated that there is no guarantee that the ensemble learning algorithms will always achieve better accuracy than a single classifier. However, this discovery can answer the questions around whether using the ensemble learning algorithm(s) will achieve better performance accuracy than single machine learning algorithm(s); however, this depends on the algorithm selected by the researchers.

**RQ. 8:** ‘When comparing the Arbiter Tree (Chan and Stolfo, 1993) and Combiner Tree (Chan and Stolfo, 1997) with the other algorithms in ensemble learning, which will achieve the better performance?’ (as indicated in Section 2.3.3)

To answer this question (Chapter 5), we chose two algorithms within ensemble learning that are commonly used in sentiment analysis. They are: Stacking (Wolpert, 1992), and Majority Voting (Polikar, 2012). Both of these were implemented using the same data and treatments as the Arbiter Tree (Chan and Stolfo, 1993) and Combiner Tree (Chan and Stolfo, 1997). In the comparison, the results from the Arbiter Tree (Chan and Stolfo, 1993) achieved better accuracy than Majority Voting (Polikar, 2012), but slightly lower than Stacking (Wolpert, 1992). However, we showed that it is worth trying to implement alternative algorithms of ensemble learning: Arbiter Tree (Chan and Stolfo, 1993) and Combiner Tree (Chan and Stolfo, 1997). Thus, we have successfully analysed, investigated and demonstrated that the Arbiter Tree (Chan and Stolfo, 1993) and Combiner Tree (Chan and Stolfo, 1997) are applicable for use to classify the sentiment of the context (Tweets and SMS).



### 6.3 Thesis Limitations

Inevitably, there are the limitations of TJP system that have not been addressed. These include features that are used in data pre-processing such as part-of-speech and stemming. However, text in Tweets is commonly short and frequently misspelled or randomly abbreviated (such as ‘prolly’ for ‘probably’). This would confuse common part-of-speech algorithms or stemming software (See e.g. Agarwal et al. (2009), Go et al. (2009), Bermingham and Smeaton (2010), Kouloumpis et al. (2011) and Saif et al. (2012)). Moreover, there is no function for using cross-validation because cross-validation could cause overfitting (Ng, 1997; Refaeilzadeh et al., 2009; Dwork et al., 2015). Overfitting normally occurs when using the training and testing data that are split from the same dataset (Ng, 1997; Refaeilzadeh et al., 2009; Dwork et al., 2015). Overfitting happens in highly tuned model that have achieved high levels good of performance at correctly classifying training data while getting less and less accurate at prediction of testing data that it has to classify (Featherstone, 2013). Therefore, a cross-validation function was not added in TJP system.

### 6.4 Summary of Contributions to knowledge of this Thesis

The achievement of the research questions described in the previous section allowed us to make three contributions to the field of sentiment analysis.

**The first contribution** is the investigation of features that are used in pre-processing data. The features concerned in this investigation were emoticons, negations, Twitter features, repeated letters, special characters, slang words and stopwords (for further details see Section 3.6). In the investigation, each feature was combined and tested with the dataset by using Naïve Bayes (Tan *et al.*, 2009). The features were applied to both training and testing data. The results revealed that using the combination of all features achieved better accuracy compared with any subset of these features.

**The second contribution** is the investigation and evaluation of the factors that may be used in sentiment analysis. These include the dataset, sentiment lexicon(s), sentiment resource(s) and machine learning algorithm(s) (Chapter 3). We explored different factors in the sentiment analysis task to find the best combination. This method used three machine learning algorithms, two sentiment resources and three sentiment lexicons. The best classification results were achieved using the combination of SVM (Kecman, 2005), training data (Wilson *et al.*, 2013), Hu and Liu's lexicons (Hu and Liu, 2004), MPQA's lexicon (Wilson *et al.*, 2005b) and AFINN's Lexicon (Nielsen, 2011a). However, with the absence of training data (Wilson *et al.*, 2013), the AFINN Lexicon (Nielsen, 2011a) achieved better accuracy performance than the other sentiment lexicons. This approach can be used to identify the factors that have an impact on sentiment analysis performance.

**The third contribution** was the investigation, implementation and evaluation of the theoretical principles through the re-contextualisation of existing techniques in ensemble learning: the Arbiter Tree (Chan and Stolfo, 1993) and Combiner Tree (Chan and Stolfo, 1997). Neither appears to have been used previously with sentiment analysis. Therefore, there are no review articles/books that describe work related to using the Arbiter Tree (Chan and Stolfo, 1993) and Combiner Tree (Chan and Stolfo, 1997) in sentiment analysis. Therefore, we intend to re-contextualise the Arbiter Tree (Chan and Stolfo, 1993) and Combiner Tree (Chan and Stolfo, 1997) to demonstrate how they can be applied to the task of sentiment analysis. Moreover, we detailed their applicability to classify the sentiment of Tweets and SMS.

## 6.5 Future Work

Our work has taken us on a journey through the world of sentiment analysis, but was limited to polarity classification. There are however several possible directions for future work emerging from the implementation of this thesis.

This thesis is based on word polarity (positive and negative). This is converted to either +1 or -1, which is discrete polarity. From this point, there is

another question that has not been investigated and answered in this thesis: ‘does it make any difference whether we use discrete and continuous polarity?’ Continuous polarity refers to a range of polarity that could be a real number (e.g. -2, -1, 0, 1, and 2) or a floating point number (e.g. 0.10, 0.15, and 0.20).

To answer this question, an unsupervised learning algorithm could be used, where the learning algorithm does not require labelled datasets as the input data. The most classic unsupervised learning method is clustering. Clustering is a set of algorithms that analyses groups of data based only on information found in the data that describes the objects and their relationships. The goal of clustering is to determine the intrinsic grouping in a set of data (Tan *et al.*, 2014). Clustering is needed for setting a threshold. A threshold is a parameter in which the upper and lower limits for the machine learning to interpret the range of polarity as positive/negative/neutral. Since the range polarity is not predefined nor should be defined by a person, clustering could provide the grouping significance of each polarity.

For example, a given range of polarity of 0 to 1, in which 0 is negative and 1 is positive. As the probability of value could be either 0 or 1, it is reasonable to put 0 to 0.49 as negative and 0.5 to 1 as positive. Nevertheless, this threshold setting does not reflect the nature of the data. Clustering, however, can group the data into clusters of range polarities and draw a threshold around the group. This could bring a negative threshold to  $< 0.3$  and positive  $> 0.3$  or negative  $< 0.7$  and positive  $> 0.7$  according to the clustering of the given dataset. The idea of clustering has been used by Maas *et al.* (2011)<sup>64</sup> to assign labels to datasets for use to classify movie reviews. The idea is that a review that has a score which is less than or equal to 4 out of 10 is negative. On the other hand, a positive review has a score which is greater than or equal to 7 out of 10 while the rest are not included in the dataset. After gathering this dataset with binary polarity, the rest of the process is conducted using a support vector machine to classify the final output.

However, to answer the question above, the datasets should have both discrete and continuous polarity, the reason being that their final prediction could be used to compare whether or not their accuracy performance is the same. The

---

<sup>64</sup> His datasets that are published only contain binary (positive and negative) polarity.

label of continuous polarity could be assigned using clustering or similar idea as Maas *et al.* (2011). An example of an ideal dataset is shown in Table 6.1. It is a sample of user reviews from TripAdvisor about Newcastle Airport Tourist Information<sup>65</sup>.

Reviews	Rate <sup>66</sup> from 1 to 5 (continuous polarity)	Ideally of discrete polarity <sup>67</sup>
Hasnt changed one bit so they dont read reviews or care. PUB ALWAYS STINKS OF VINEGAR..... if you dont wash and clean a bar properly ROTTEN BEER WILL STINK OF VINEGAR. Beer is warm and undrinkable	1	-1 (negative)
This airport needs to move into the 21st century, all very well having planes going all over the world but when you get back it is a bit ridiculous to wait nearly an hour for luggage. This ruins what was a great holiday being tired already after nearly 22 hours total travelling. Always convenient to get to and the flying out is very good but the coming back part is the letdown.	3	-1 (negative)
The first time I've flown from Newcastle for a few years. Everything went as planned the new bars and eateries upstairs were fine if not a little rowdy (stag and hen parties) that cant be helped.	4	1 (positive)
I had a great experience at the cabin, the staff were great and couldnt have been more helpful :) i would definatly choose the cabin again.	5	1 (positive)

**Table 6-1:** Example of dataset that has both discrete and continuous polarity

However, another process that could be used to answer this question is ‘meta-analysis’. Meta-analysis is a process that compares and combines quantitative results from several studies in the same area using statistics. By using meta-analysis, as many works as possible that are related to discrete and continuous polarity should be collected. The hypothesis for discrete and continuous polarity should be set as; there is no difference between using either discrete or continuous polarity. After that, the statistical method will be used to

<sup>65</sup> [http://www.tripadvisor.co.uk/Attraction\\_Review-g186394-d213735-Reviews-Newcastle\\_Airport\\_Tourist\\_Information-Newcastle\\_upon\\_Tyne\\_Tyne\\_and\\_Wear\\_England.html](http://www.tripadvisor.co.uk/Attraction_Review-g186394-d213735-Reviews-Newcastle_Airport_Tourist_Information-Newcastle_upon_Tyne_Tyne_and_Wear_England.html)

<sup>66</sup> Based on the user’s rate in the website

<sup>67</sup> This ideally is assigned manually by human

prove this hypotheses and their significance. A guide for choosing the appropriate statistics is presented in Figure 5.13.

Besides discrete and continuous polarity, there is another question of interest: will the accuracy improve when using a combination of sentiment classification and subjective classification? It has been observed in several studies that subjectivity classification may help to improve the performance of sentiment analysis. However, experiments conducted by Esuli and Sebastiani (2006a) and Zagibalov (2010) concluded that sentiment classification and subjective classification are separate tasks that simultaneously have to deal with a mixture of objective and subjective documents. This suggestion is led from sub-topics within sentiment analysis; they are sentiment classification and subjective classification. Sentiment classification is the task that classifies opinionated contexts as expressing a positive or a negative. On the other hand, subjective classification is a task that classifies a context as subjective or objective. Subjective refers to the opinion that expressions describe people's sentiments or feelings toward entities (Liu, 2010). Objective concerns entities, events and their properties (Liu, 2010). This may be relevant to our work as our sentiment analysis focuses on both positive and negative contexts. Neutral sentiment tends to be much harder to identify as it requires the determination of the contexts of the message; for example, some content may have both subjective and objective senses. Handling these contents will therefore require the introduction of another classifier to identify the subjective and objective contexts.

In addition, there can be mixed sentiment contents. Many studies did not include mixed sentiment contents in the task due to the complexity of the ambiguously defined and typically inconsistent labelling (Bermingham, 2011). However, this does not mean that the mixed sentiment contents do not exist in the real-world. This task still remains for future work to identify how the mixed sentiment contents can be better identified using machine learning algorithms. Mixed sentiment content refers to the contents that have both positive and negative sentiments.

Nevertheless, content alone is inadequate for sentiment analysis. Humans use sociocultural data to interpret meanings from a piece of information. The most

obvious examples are sarcasm and persuasion. In order to understand sarcasm and persuasion in a microblog post, people use a combination of knowledge, experience and the history of interactions between different parties as the context.

However, to locate documents on a continuum, stretching from the extremely negative to the extremely positive is still a problem. Experiments in extreme polarity areas would require a special corpus that can be used to test the accuracy of the contents of a sentiment analysis. The corpus must follow the dimensional paradigm. It must use a specialised annotation scheme, which also needs a significant research effort with future work.

Another suggestion for further research is the real-time sentiment application for analysing some social networks such as Twitter and Facebook. The question arises: ‘Is it possible for using real-time sentiment application to detect review from the customers?’ This application will be useful for companies that are interested in how their customers perceive their products or services. Moreover, a language-independent approach would make it possible to monitor different national markets, while the absence of domain-dependency would allow a system to follow the twists of language use that occurs in real-life human communication. For example, the emerging of new topics of conversation with different styles of phrasing, speech and language are those which are difficult to predict.

There is another question that not answered in this thesis, which is ‘whether the number of rules effects the improvement of performance accuracy in Arbiter Tree (Chan and Stolfo, 1993)?’ To answer this question, each rule and each pairs should be tested independently. After that, their results could be compared for finding their effective and using for further study.

## References

- ABDEL-DAYEM, A. R. 2010. Proceedings of the 7th international conference on Image Analysis and Recognition - Volume Part II, Portugal. 2177026: Springer-Verlag, pp: 120-130.
- ABDUL-MAGEED, M. and DIAB, M. 2014. Proceedings of the Language Resources and Evaluation Conference (LREC), Reykjavik, Iceland. pp: 1162-1169.
- AGARWAL, A., BIADSY, F. and MCKEOWN, K. R. 2009. Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics, Athens, Greece. 1609069: Association for Computational Linguistics, pp: 24-32.
- AHMAD, K. and ALMAS, Y. 2005. the 9th International Conference on Information Visualisation. pp: 363-368.
- AISOPOS, F., PAPADAKIS, G. and VARVARIGOU, T. 2011. Proceedings of the 3rd ACM the Special Interest Group on Multimedia (SIGMM) international workshop on Social media, Scottsdale, Arizona, USA. 2072614: ACM, pp: 9-14.
- ALM, C. O., ROTH, D. and SPROAT, R. 2005. Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing. Association for Computational Linguistics, pp: 579-586.
- ALTMAN, N. S. 1992. An introduction to kernel and nearest-neighbor nonparametric regression. *The American Statistician*, Vol. 46, pp. 175-185.
- AMAN, S. and SZPAKOWICZ, S. 2007. Text, Speech and Dialogue. Springer, pp: 196-205.
- AMBATI, V. 2008. Adv. MT Seminar Course Report. pp.
- AMIRI, H. and CHUA, T.-S. 2012. Sentiment Classification Using the Meaning of Words. In: JANNACH, D., ANAND, S. S., MOBASHER, B. and KOBZA, A. (eds.) *Intelligent Techniques for Web Personalization and Recommender Systems: AAAI Technical Report WS-12-09*. Palo Alto, California: The AAAI Press, Isbn: 9781577355748.
- AUE, A. and GAMON, M. 2005. Proceedings of recent advances in natural language processing (RANLP). pp: 2-1.
- BACCIANELLA, S., ESULI, A. and SEBASTIANI, F. 2010a. *SentiWordNet*. Available: <http://sentiwordnet.isti.cnr.it/>
- BACCIANELLA, S., ESULI, A. and SEBASTIANI, F. 2010b. Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10), Valletta, Malta. European Language Resources Association (ELRA), pp.
- BALAHUR, A. 2013. 4th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis, Atlanta, Georgia. Association for Computational Linguistics, pp: 120-128.
- BALDWIN, R. A. 2009. Use of maximum entropy modeling in wildlife research. *Entropy*, Vol. 11, pp. 854-866.
- BALLAN, L., BERTINI, M., BIMBO, A., SEIDENARI, L. and SERRA, G. 2011. Event detection and recognition for semantic annotation of video. *Multimedia Tools and Applications*, Vol. 51, pp. 279-302.
- BARAN, E. and WARRY, F. 2008. Statistical tests for comparing samples. *Simple data analysis for biologists*. WorldFish, Isbn: 9789995071011.
- BARTLETT, M. S., LITTLEWORT, G., FRANK, M., LAINSCSEK, C., FASEL, I. and MOVELLAN, J. 2005. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR). pp: 568-573.

- BATTOCCHI, A., PIANESI, F. and GOREN-BAR, D. 2005. Dafex: Database of facial expressions. *Intelligent Technologies for Interactive Entertainment*. Springer, Isbn: 3540305092.
- BEAR. 2015. *Great Debut Film Performance By Eminem!* [Online]. <http://www.amazon.co.uk/>. Available: [http://www.amazon.co.uk/product-reviews/B00006FMGR/ref=cm\\_cr\\_pr\\_hist\\_5?ie=UTF8&filterBy=addFiveStar&howViewpoints=0&sortBy=bySubmissionDateDescending](http://www.amazon.co.uk/product-reviews/B00006FMGR/ref=cm_cr_pr_hist_5?ie=UTF8&filterBy=addFiveStar&howViewpoints=0&sortBy=bySubmissionDateDescending).
- BEHNEL, S., BRADSHAW, R., CITRO, C., DALCIN, L., SELJEBOTN, D. S. and SMITH, K. 2011. Cython: The best of both worlds. *Computing in Science & Engineering*, Vol. 13, pp. 31-39.
- BEHNEL, S., BRADSHAW, R. and SELJEBOTN, D. 2008. Cython: C-extensions for Python.
- BERMEJO, P., GÁMEZ, J. A. and PUERTA, J. M. 2011. Improving the performance of Naive Bayes multinomial in e-mail foldering by introducing distribution-based balance of datasets. *Expert Systems with Applications*, Vol. 38, pp. 2072-2080.
- BERMINGHAM, A. 2011. *Sentiment analysis and real-time microblog search*. Dublin City University.
- BERMINGHAM, A. and SMEATON, A. F. 2011. Proceedings of the Workshop on Sentiment Analysis where AI meets Psychology (SAAIP 2011). Asian Federation of Natural Language Processing, pp: 2-10.
- BIFET, A. and FRANK, E. 2010. Proceedings of the 13th international conference on Discovery science, Canberra, Australia. Springer-Verlag, pp: 1-15.
- BIKEL, D. M., SCHWARTZ, R. and WEISCHEDEL, R. M. 1999. An algorithm that learns what's in a name. *Machine learning*, Vol. 34, pp. 211-231.
- BIRD, S. 2006a. Proceedings of the COLING/ACL on Interactive presentation sessions. Association for Computational Linguistics, pp: 69-72.
- BIRD, S. 2006b. Proceedings of the COLING/ACL on Interactive presentation sessions. Association for Computational Linguistics, pp: 69-72.
- BIRD, S., KLEIN, E. and LOPER, E. 2009a. Accessing Text Corpora and Lexical Resources. *Natural Language Processing with Python*. O'Reilly Media, Isbn: 9780596516499
- BIRD, S., KLEIN, E. and LOPER, E. 2009b. *Natural language processing with Python*, O'Reilly, Isbn: 9780596516499.
- BISHOP, C. M. 2006. *Pattern recognition and machine learning*, springer New York, Isbn: 9780387310732.
- BLACK, K. 2011. Sampling and Sampling Distribution. *Business Statistics: For Contemporary Decision Making*. Wiley, Isbn: 9780470931462.
- BLEI, D. M., NG, A. Y. and JORDAN, M. I. 2003. Latent dirichlet allocation. *the Journal of machine Learning research*, Vol. 3, pp. 993-1022.
- BLITZER, J., DREDZE, M. and PEREIRA, F. 2007. ACL. pp: 440-447.
- BRADLEY, M. M. and LANG, P. J. 1999. Affective norms for English words (ANEW): Instruction manual and affective ratings. Technical Report C-1, The Center for Research in Psychophysiology, University of Florida.
- BREIMAN, L. 1996. Bagging predictors. *Machine learning*, Vol. 24, pp. 123-140.
- BREIMAN, L. 2001. Random forests. *Machine learning*, Vol. 45, pp. 5-32.
- BRILL, E. 1994. Proceedings of the twelfth national conference on Artificial intelligence, Seattle, Washington, USA. 199378: American Association for Artificial Intelligence, pp: 722-727.
- BUTLER, E. 2010. Senior Seminar Conference, University of Minnesota, Morris. pp: 11-16.
- CARIDAKIS, G., WAGNER, J., RAOUZAIYOU, A., CURTO, Z., ANDRÉ, E. and KARPOUZIS, K. 2010. Multimodal Corpora: Advances in Capturing, Coding and Analyzing Multimodality. pp: 80.



- CELIKILMAZ, A., HAKKANI-TUR, D. and FENG, J. 2010. Spoken Language Technology Workshop (SLT). IEEE, pp: 79-84.
- CHAKRABORTY, G., PAGOLU, M. and GARLA, S. 2014. Clustering and Topic Extraction. *Text Mining and Analysis: Practical Methods, Examples, and Case Studies Using SAS*. SAS Institute, ISBN: 9781612907871.
- CHALOTHORN, T. and ELLMAN, J. 2014. Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014), Dublin, Ireland. Association for Computational Linguistics and Dublin City University, pp.
- CHAN, P. K. 1996. *An extensible meta-learning approach for scalable and accurate inductive learning*. Columbia University
- CHAN, P. K. and STOLFO, S. J. 1993. The International Association for the Advancement of Artificial Intelligence (AAAI) workshop in Knowledge Discovery in Databases. pp: 227-240.
- CHAN, P. K. and STOLFO, S. J. 1995. Conference on Knowledge Discovery and Data Mining (KDD). pp: 39-44.
- CHAN, P. K. and STOLFO, S. J. 1997. On the accuracy of meta-learning for scalable data mining. *Journal of Intelligent Information Systems*, Vol. 8, pp. 5-28.
- CHANG, C.-C. and LIN, C.-J. 2011. LIBSVM: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, Vol. 2, pp. 27.
- CHATFIELD, C. 1983a. The design and analysis of experiments - 1 Comparative experiments. *Statistics for Technology: A Course in Applied Statistics, Third Edition*. Taylor & Francis, ISBN: 9780412253409.
- CHATFIELD, C. 1983b. The design and analysis of experiments - 2 Factorial experiments. *Statistics for Technology: A Course in Applied Statistics, Third Edition*. Taylor & Francis, ISBN: 9780412253409.
- CHEN, J. Y. and LONARDI, S. 2009. *Biological Data Mining*, Taylor & Francis, ISBN: 9781420086843.
- CHEN, T. and KAN, M.-Y. 2013. Creating a live, public short message service corpus: the NUS SMS corpus. *Language Resources and Evaluation*, Vol. 47, pp. 299-335.
- CHOW, S.-C. and LIU, J.-P. 2004. Designs for Clinical Trials. *Design and Analysis of Clinical Trials: Concepts and Methodologies*. Wiley, ISBN: 9780471249856.
- CHOWDHURY, G. G. 2010. Natural language processing and information retrieval. *Introduction to Modern Information Retrieval*. Neal-Schuman Publishers, ISBN: 9781856046947.
- CHOY, M., CHEONG, M. L., LAIK, M. N. and SHUNG, K. P. 2011. A sentiment analysis of Singapore Presidential Election 2011 using Twitter data with census correction. *arXiv preprint arXiv:1108.5520*, Vol.
- CIPRA, B. A. 1987. An introduction to the Ising model. *American Mathematical Monthly*, Vol. 94, pp. 937-959.
- COHEN, J. 1968. Weighted kappa: Nominal scale agreement provision for scaled disagreement or partial credit. *Psychological bulletin*, Vol. 70, pp. 213.
- CORRIGAN, J. 2008. *The Oxford Handbook of Religion and Emotion*, Oxford University Press, USA, ISBN: 9780195170214.
- CRAMMER, K. and SINGER, Y. 2003. Ultraconservative online algorithms for multiclass problems. *The Journal of Machine Learning Research*, Vol. 3, pp. 951-991.
- CRISTIANINI, N. and SHAWE-TAYLOR, J. 2000. *An introduction to support vector machines and other kernel-based learning methods*, Cambridge university press, ISBN: 0521780195.
- CUFOGLU, A., LOHI, M. and MADANI, K. 2008. International Conference on Computer Engineering & Systems (ICCES). pp: 210-215.

- CUNNINGHAM, H. 2002. Proceedings of the 40th Annual Meeting on Association for Computational Linguistic. pp: 168-175.
- CUNNINGHAM, H., MAYNARD, D. and BONTCHEVA, K. 2011. *Text Processing with Gate (Version 6)*, Gate, Isbn: 9780956599315.
- DANET, B., RUEDENBERG-WRIGHT, L. and ROSENBAUM-TAMARI, Y. 1997. "HMMM...WHERE'S THAT SMOKE COMING FROM?". *Journal of Computer-Mediated Communication*, Vol. 2.
- DASGUPTA, S., KALAI, A. T. and MONTELEONI, C. 2009. Analysis of perceptron-based active learning. *The Journal of Machine Learning Research*, Vol. 10, pp. 281-299.
- DAY, D., MCHENRY, C., KOZIEROK, R. and RIEK, L. 2004. International Conference on Language Resources and Evaluation. pp.
- DENECKE, K. 2008. IEEE 24th International Conference Data Engineering Workshop (ICDEW) pp: 507-512.
- DERCZYNSKI, L. 2013. *Natural Language Toolkit: SVM-based classifier*. Available: <http://www.nltk.org/modules/nltk/classify/svm.html>
- DERKS, D., FISCHER, A. H. and BOS, A. E. 2008. The role of emotion in computer-mediated communication: A review. *Computers in Human Behavior*, Vol. 24, pp. 766-785.
- DEVITT, A. and AHMAD, K. 2007. Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics. Association for Computational Linguistics, pp: 984-991.
- DI EUGENIO, B. and GLASS, M. 2004. The kappa statistic: A second look. *Computational linguistics*, Vol. 30, pp. 95-101.
- DOMHOFF, G. W. 2003. *The Scientific Study of Dreams: Neural Networks, Cognitive Development, and Content Analysis*, Amer Psychological Assn, Isbn: 9781557989352.
- DONG, Z. and DONG, Q. 2006. *Hownet And the Computation of Meaning*, World Scientific Publishing Co., Inc., Isbn: 9812564918.
- DONNE, J. 2013. *Delphi Complete Poetical Works of John Donne (Illustrated)*, Delphi Classics, Isbn: 9781908909763.
- DONNE, J. and ALFORD, H. 1839. *The Works of John Donne: With a Memoir of His Life*, Parker, Isbn: -.
- DOWNEY, A. B. 2014. Relationships Between Variables. *Think Stats*. O'Reilly Media, Isbn: 9781491907375.
- DU, S. 2008. *On the Use of Natural Language Processing for Automated Conceptual Data Modeling*. University of Pittsburgh.
- DUAN, W., CAO, Q., YU, Y. and LEVY, S. 2013. System Sciences (HICSS), 2013 46th Hawaii International Conference on. IEEE, pp: 3119-3128.
- DWORK, C., FELDMAN, V., HARDT, M., PITASSI, T., REINGOLD, O. and ROTH, A. 2015. Advances in Neural Information Processing Systems. pp: 2341-2349.
- EKMAN, P. and FRIESEN, W. 1978. Facial Action Coding System: A Technique for the Measurement of Facial Movement. *Consulting Psychologists Press*, Vol. -.
- ELANGOVAN, M., RAMACHANDRAN, K. I. and SUGUMARAN, V. 2010. Studies on Bayes classifier for condition monitoring of single point carbide tipped tool based on statistical and histogram features. *Expert Systems with Applications*, Vol. 37, pp. 2059-2065.
- ELLIOTT, R. J., AGGOUN, L. and MOORE, J. B. 1995. *Hidden Markov Models*, Springer, Isbn: 9780387943640.
- ESULI, A. and SEBASTIANI, F. 2005. Proceedings of the 14th ACM international conference on Information and knowledge management, Bremen, Germany. ACM, pp: 617-624.

- 
- ESULI, A. and SEBASTIANI, F. 2006a. Proceedings of EACL-06, 11th Conference of the European Chapter of the Association for Computational Linguistics, Trento, Italy. pp: 193-200.
- ESULI, A. and SEBASTIANI, F. 2006b. In Proceedings of the 5th Conference on Language Resources and Evaluation (LREC'06), Genoa, Italy. pp: 417-422.
- ESULI, A. and SEBASTIANI, F. 2007. SENTIWORDNET: A high-coverage lexical resource for opinion mining. Technical Report 2007-TR-02, Istituto di Scienza e Tecnologie dell'Informazione, Consiglio Nazionale delle Ricerche, Pisa, IT.
- EVANS, B. I. 2013. *The Language of Shakespeare's Plays*, Taylor & Francis, Isbn: 9781136560699.
- FAN, R.-E., CHANG, K.-W., HSIEH, C.-J., WANG, X.-R. and LIN, C.-J. 2008. LIBLINEAR: A library for large linear classification. *The Journal of Machine Learning Research*, Vol. 9, pp. 1871-1874.
- FATTAH, H. M. 2006. Growing Unarmed Battalion in Qaeda Army Is Using Internet to Get the Message Out. *30 September 2006*. <http://www.nytimes.com/>: The New York Times.
- FEATHERSTONE, C. 2013. Adaptive Science and Technology (ICAST), 2013 International Conference on. IEEE, pp: 1-8.
- FELLBAUM, C. 1998. *WordNet: An Electronic Lexical Database*, Mit Press, Isbn: 9780262061971.
- FELLBAUM, C. 2010. Wordnet. In: POLI, R., HEALY, M. and KAMEAS, A. (eds.) *Theory and applications of ontology: computer applications*. Springer, Isbn: 9789048188475.
- FERNÁNDEZ-DELGADO, M., CERNADAS, E., BARRO, S. and AMORIM, D. 2014. Do we need hundreds of classifiers to solve real world classification problems? *The Journal of Machine Learning Research*, Vol. 15, pp. 3133-3181.
- FIELD, A. 2013a. Everything you ever wanted to know about statistics (well, sort of). *Discovering statistics using IBM SPSS statistics*. Sage, Isbn: 9781446274583.
- FIELD, A. 2013b. Exploring Assumptions. *Discovering statistics using IBM SPSS statistics*. Sage, Isbn: 9781446274583.
- FIELD, A. 2013c. Non-parametric tests. *Discovering statistics using IBM SPSS statistics*. Sage, Isbn: 9781446274583.
- FREUND, Y. and SCHAPIRE, R. E. 1996. International Conference on Machine Learning (ICML). pp: 148-156.
- GALLAVOTTI, G. 1972. Instabilities and phase transitions in the Ising model. A review. *La Rivista del Nuovo Cimento*, Vol. 2, pp. 133-169.
- GAURAV, M., SRIVASTAVA, A., KUMAR, A. and MILLER, S. 2013. Proceedings of the 7th Workshop on Social Network Mining and Analysis. ACM, pp: 7.
- GAUTAM, G. and YADAV, D. 2014. Contemporary Computing (IC3), 2014 Seventh International Conference on. IEEE, pp: 437-442.
- GERSHENSON, C. 2003. Artificial neural networks for beginners. *arXiv preprint cs/0308031*, Vol.
- GHAHRAMANI, Z. 2001. An introduction to hidden Markov models and Bayesian networks. *International Journal of Pattern Recognition and Artificial Intelligence*, Vol. 15, pp. 9-42.
- GHORBEL, H. and JACOT, D. 2011. Sentiment analysis of French movie reviews. *Advances in Distributed Agent-Based Retrieval Tools*, Vol., pp. 97-108.
- GIANNAKOPOULOS, G. and KARKALETSIS, V. 2009. Proceedings of Text Analysis Conference TAC 2009, Maryland, USA. pp: 1.
- GILDEA, D. and JURAFSKY, D. 2002. Automatic labeling of semantic roles. *Computational linguistics*, Vol. 28, pp. 245-288.
- GIMÉNEZ, J. and AMIGÓ, E. 2006. The 5th International Conference on Language Resources and Evaluation (LREC'06). pp.
-

- 
- GO, A., BHAYANI, R. and HUANG, L. 2009. Twitter sentiment classification using distant supervision. *CS224N Natural Language Processing, Project Report, Stanford*, Vol. 1, pp. 12.
- GOODWIN, C. J. 2009. Experimental Design II: Factorial Designs. *Research In Psychology: Methods and Design*. John Wiley & Sons, Isbn: 9780470522783.
- GOSLING, J. 1995. Statistical inference - hypothesis testing. *Introductory Statistics*. Pascal Press, Isbn: 9781864410150.
- GRABNER, D., ZANKER, M., FLIEDL, G. and FUCHS, M. 2012. 19th Conference on Information and Communication Technologies in Tourism (ENTER), Helsingborg, Sweden. Springer-Verlag Wien, pp.
- GRAEPEL, T., CANDELA, J. Q., BORCHERT, T. and HERBRICH, R. 2010. Proceedings of the 27th International Conference on Machine Learning (ICML-10). pp: 13-20.
- GRANT, C. and TOMAL, D. R. 2013. Presenting your results and discussion in the final manuscript. *How to Finish and Defend Your Dissertation : Strategies to Complete the Professional Practice Doctorate*. Lanham: R&L Education, Isbn: 9781475804027.
- GRYC, W. and MOILANEN, K. 2014. Leveraging Textual Sentiment Analysis with Social Network Modelling. *From Text to Political Positions: Text analysis across disciplines*, Vol. 55, pp. 47.
- GUERINI, M., GATTI, L. and TURCHI, M. 2013. Tthe 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP'13), Seattle, Washington, USA. pp: 1259-1269.
- HALL, M., FRANK, E., HOLMES, G., PFAHRINGER, B., REUTEMANN, P. and WITTEN, I. H. 2009. The WEKA data mining software: an update. *ACM SIGKDD explorations newsletter*, Vol. 11, pp. 10-18.
- HANAND, E. and KARYPIS, G. 2000. European Conference on Principles of Data Mining and Knowledge Discovery. pp: 424-431.
- HARTE, J. 2011. *Maximum entropy and ecology: a theory of abundance, distribution, and energetics*, Oxford University Press, Isbn: 9780191621161.
- HARUECHAIYASAK, C., KONGTHON, A., PALINGOON, P. and SANGKEETTRAKARN, C. 2010. Proceedings of the 8th Workshop on Asian Language Resouces, Beijing, China. Coling 2010 Organizing Committee, pp: 64-71.
- HATZIVASSILOGLOU, V. and MCKEOWN, K. R. 1997a. Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics and Eighth Conference of the European Chapter of the Association for Computational Linguistics, Madrid, Spain. 979640: Association for Computational Linguistics, pp: 174-181.
- HATZIVASSILOGLOU, V. and MCKEOWN, K. R. 1997b. Proceedings of the 35th annual meeting of the association for computational linguistics and eighth conference of the european chapter of the association for computational linguistics. Association for Computational Linguistics, pp: 174-181.
- HATZIVASSILOGLOU, V. and WIEBE, J. M. 2000. Proceedings of the 18th conference on Computational linguistics-Volume 1. Association for Computational Linguistics, pp: 299-305.
- HAYES, B. E. 2008. Determing Customer Requirements. *Measuring Customer Satisfaction and Loyalty: Survey Design, Use, and Statistical Analysis Methods*. ASQ Quality Press, Isbn: 9780873897433.
- HEALEY, J. 2012. Introduction to inferential statistics, the sampling distribution, and estimation. *The Essentials of Statistics: A Tool for Social Research*. Cengage Learning, Isbn: 9781133713586.
-

- 
- HEARST, M. A., DUMAIS, S. T., OSMAN, E., PLATT, J. and SCHOLKOPF, B. 1998. Support vector machines. *IEEE Intelligent Systems and their Applications*, Vol. 13, pp. 18-28.
- HLIAOUTAKIS, A. and PETRAKIS, E. M. 2011. Automatic Term Identification by User Profile for Document Categorisation in Medline. In: MUÑOZ, R., MONTOYO, A. and MÉTAIS, E. (eds.) *Natural Language Processing and Information Systems*. Springer Berlin Heidelberg, Isbn: 9783642223266.
- HO, T. K. 1998. The random subspace method for constructing decision forests. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 20, pp. 832-844.
- HOBSON, J. A., STICKGOLD, R. and PACE-SCHOTT, E. F. 1998. The neuropsychology of REM sleep dreaming. *NeuroReport*, Vol. 9, pp. 1-14.
- HOGENBOOM, A., BOON, F. and FRASINCAR, F. 2012. A Statistical Approach to Star Rating Classification of Sentiment. In: CASILLAS, J., MARTÍNEZ-LÓPEZ, F. J. and CORCHADO RODRÍGUEZ, J. M. (eds.) *Management Intelligent Systems*. Springer Berlin Heidelberg, Isbn: 9783642308635.
- HOLMES, G., DONKIN, A. and WITTEN, I. H. 1994. Proceedings of the 2nd Australian and New Zealand Conference on Intelligent Information Systems. IEEE, pp: 357-361.
- HOPE, C. 2009. Home Office fails to shut down a single extremist website in two years. 19 March 2009. <http://www.telegraph.co.uk/>; The Telegraph.
- HOPE, L. R. and KORB, K. B. 2004. Proceedings of the 17th Australian joint conference on Advances in Artificial Intelligence, Cairns, Australia. 2146938: Springer-Verlag, pp: 991-997.
- HOSMER JR, D. W., LEMESHOW, S. and STURDIVANT, R. X. 2013. *Applied logistic regression*, John Wiley & Sons, Isbn: 9781118548394.
- HOWITT, D. and CRAMER, D. 2003. Numerical ways of describing relationships Correlation coefficients. *First Steps In Research and Statistics: A Practical Workbook for Psychology Students*. Taylor & Francis, Isbn: 9781134635498.
- HU, M. and LIU, B. 2004. Proceedings of the tenth ACM Special Interest Group on Knowledge Discovery and Data Mining (SIGKDD) international conference on Knowledge discovery and data mining, Seattle, WA, USA. 1014073: ACM, pp: 168-177.
- HU, M. and LIU, B. 2006. AAAI Spring Symposium: Computational Approaches to Analyzing Weblogs. pp: 61-66.
- HU, N., BOSE, I., KOH, N. S. and LIU, L. 2012. Manipulation of online reviews: An analysis of ratings, readability, and sentiments. *Decision Support Systems*, Vol. 52, pp. 674-684.
- IBM. 2010. SPSS.
- INFOGISTICS LTD. 2000. *NLProcessor - Text Analysis Toolkit*. Available: <http://www.infogistics.com/textanalysis.html>
- INTERNATIONAL ORGANIZATION FOR STANDARDIZATION 1986. ISO 8879:1986. 23 October 1986. <https://www.iso.org>; ISO, Geneva.
- IZARD, C. E. 1991. *The psychology of emotions*, Springer Science & Business Media, Isbn: 0306438658.
- JAIN, J. and JAIN, A. 1981. Displacement Measurement and Its Application in Interframe Image Coding. *IEEE Transactions on Communications*, Vol. 29, pp. 1799-1808.
- JAYNES, E. T. 1957. Information theory and statistical mechanics. *Physical review*, Vol. 106, pp. 620.
- JIN, X., LI, Y., MAH, T. and TONG, J. 2007. Proceedings of the 1st international workshop on Data mining and audience intelligence for advertising, San Jose, California. 1348604: ACM, pp: 28-33.
-

- 
- JMAL, J. and FAIZ, R. 2013. Proceedings of the 3rd International Conference on Web Intelligence, Mining and Semantics. ACM, pp: 33.
- JOACHIMS, T. 2002a. *Learning to classify text using support vector machines: Methods, theory and algorithms*, Kluwer Academic Publishers, Isbn: 9780792376798.
- JOACHIMS, T. 2002b. Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, pp: 133-142.
- JONES, E., OLIPHANT, T. and PETERSON, P. 2001. *SciPy: Open source scientific tools for Python*.
- JOSE. 2015. *Too long a queue* [Online]. <http://www.tripadvisor.co.uk/>. Available: [http://www.tripadvisor.co.uk/ShowUserReviews-g186338-d553603-r277789844-The\\_London\\_Eye-London\\_England.html#CHECK\\_RATES\\_CONT](http://www.tripadvisor.co.uk/ShowUserReviews-g186338-d553603-r277789844-The_London_Eye-London_England.html#CHECK_RATES_CONT).
- JUNGHER, A., JÜRGENS, P. and SCHOEN, H. 2012. Why the pirate party won the german election of 2009 or the trouble with predictions: A response to tumasjan, a., sprenger, to, sander, pg, & welp, im “predicting elections with twitter: What 140 characters reveal about political sentiment”. *Social science computer review*, Vol. 30, pp. 229-234.
- KARLGREN, J., SAHLGREN, M., OLSSON, F., ESPINOZA, F. and HAMFORS, O. 2012. Usefulness of Sentiment Analysis. In: BAEZA-YATES, R., VRIES, A., ZARAGOZA, H., CAMBAZOGLU, B. B., MURDOCK, V., LEMPEL, R. and SILVESTRI, F. (eds.) *Advances in Information Retrieval*. Springer Berlin Heidelberg, Isbn: 9783642289965.
- KEARNS, M. 1988. Thoughts on hypothesis boosting. *Unpublished manuscript*, Vol. 45, pp. 105.
- KECMAN, V. 2005. Support Vector Machines – An Introduction. In: WANG, L. (ed.) *Support Vector Machines: theory and applications*. Springer Science & Business Media, Isbn: 9783540243885.
- KESHTKAR, F. and INKPEN, D. 2010. Proceedings of the North American Chapter of the Association for Computational Linguistics - Human Language Technologies (NAACL HLT): Workshop on Computational Approaches to Analysis and Generation of Emotion in Text, Los Angeles, California. 1860636: Association for Computational Linguistics, pp: 35-44.
- KIEU, B. T. and PHAM, S. B. 2010. 2nd International Conference on Knowledge and Systems Engineering (KSE). pp: 152-157.
- KIM, S. M. and CALVO, R. A. 2011. Sentiment-Oriented Summarisation of Peer Reviews. Springer Berlin / Heidelberg, Isbn: 9783642218682.
- KIM, S. M. and HOVY, E. 2004. Proceedings of the 20th international conference on Computational Linguistics, Geneva, Switzerland. Association for Computational Linguistics, pp.
- KLEINBAUM, D. G. and KLEIN, M. 2010. *Logistic regression: a self-learning text*, Springer, Isbn: 9781441917423.
- KOEHN, P. 2010. Moses, statistical machine translation system, user manual and code guide. University of Edinburgh.
- KOEHN, P., HOANG, H., BIRCH, A., CALLISON-BURCH, C., FEDERICO, M., BERTOLDI, N., COWAN, B., SHEN, W., MORAN, C., ZENS, R., DYER, C., BOJAR, O., CONSTANTIN, A. and HERBST, E. 2007. Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions, Prague, Czech Republic. 1557821: Association for Computational Linguistics, pp: 177-180.
- KOGA, T., IINUMA, K., HIRANO, A., IJIMA, Y. and ISHIGURO, T. 1981. Proceedings of national Telecommunications conference, New Orleans, LA. pp: G5.3.1–G5.3.5.
-



- KOLYAL, A. K., EKBAL, A. and BANDYOPADHYAY, S. 2013. Using Voting Approach for Event Extraction and Event-DCT, Event-Time Relation Identification. *International Journal*, Vol. 4, pp. 65.
- KONGTHON, A., HARUECHAIYASAK, C., SANGKEETTRAKARN, C., PALINGOON, P. and WUNNASRI, W. 2011. Proceedings of Technology Management in the Energy Smart World (PICMET). pp: 1-6.
- KÖNIG, A. C. and BRILL, E. 2006. Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, pp: 598-603.
- KOTSIANTIS, S. B., ZAHARAKIS, I. and PINTELAS, P. 2007. Proceedings of the 2007 conference on Emerging Artificial Intelligence Applications in Computer Engineering: Real Word AI Systems with Applications in eHealth, HCI, Information Retrieval and Pervasive Technologies, Amsterdam, The Netherlands. pp: 3-24.
- KOULOUMPIS, E., WILSON, T. and MOORE, J. 2011. Proceedings of the 5th International Association for the Advancement of Artificial Intelligence (AAAI) Conference on Weblogs and Social Media, Barcelona, Spain. pp.
- KRIPPENDORFF, K. 2011. Agreement and information in the reliability of coding. *Communication Methods and Measures*, Vol. 5, pp. 93-112.
- KRIPPENDORFF, K. H. 1980. *Content analysis: an introduction to its methodology*, Beverly Hills, California, Calif Sage Publications, Isbn: 9780803914988.
- KU, L. W., HUANG, T. H. and CHEN, H. H. 2009. Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing:, Singapore. 1699672: Association for Computational Linguistics, pp: 1260-1269.
- KÜBLBECK, C. and ERNST, A. 2006. Face detection and tracking in video sequences using the modifiedcensus transformation. *Image and Vision Computing*, Vol. 24, pp. 564-572.
- KUSHWAHA, K. S. 2009. Chi-Square Distribution and Its Application (Or Chi-Square Statistic). *Theory of Sample Surveys and Statistical Decisions*. New India Publishing Agency, Isbn: 9788189422899.
- LAFFERTY, J., MCCALLUM, A. and PEREIRA, F. C. 2001. Proceedings of the Eighteenth International Conference on Machine Learning (ICML'01), San Francisco, CA, USA. Morgan Kaufmann Publishers Inc., pp: 282-289.
- LANGVILLE, A. N., MEYER, C. D. and FERNÁNDEZ, P. 2008. Google's pagerank and beyond: The science of search engine rankings. *The Mathematical Intelligencer*, Vol. 30, pp. 68-69.
- LE, Z. 2004. Maximum entropy modeling toolkit for Python and C++. Natural Language Processing Lab, Northeastern University, China.
- LEUBA, C. J. 1961. *Man: A General Psychology*, Holt, Rinehart and Winston, Isbn.
- LEVINE, D. M. and STEPHAN, D. F. 2009. Sampling Distributions and Confidence Intervals. *Even You Can Learn Statistics: A Guide for Everyone Who Has Ever Been Afraid of Statistics*. Pearson Education, Isbn: 9780137025930.
- LEWIS, D. D. 1998. Proceedings of the 10th European Conference on Machine Learning. 649711: Springer-Verlag, pp: 4-15.
- LIANG, P., BOUCHARD-CÔTÉ, A., KLEIN, D. and TASKAR, B. 2006. Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics. Association for Computational Linguistics, pp: 761-768.
- LIANGXIAO, J., ZHANG, H. and ZHIHUA, C. 2009. A Novel Bayes Model: Hidden Naive Bayes. *IEEE Transactions on Knowledge and Data Engineering*, Vol. 21, pp. 1361-1371.

- LIN, C. J., WENG, R. C. and KEERTHI, S. S. 2008. Trust region newton method for logistic regression. *The Journal of Machine Learning Research*, Vol. 9, pp. 627-650.
- LINGENFELSER, F., WAGNER, J. and ANDRÉ, E. 2011. Proceedings of the 13th international conference on multimodal interfaces. ACM, pp: 19-26.
- LIU, B. 2007. *Web data mining: exploring hyperlinks, contents, and usage data*, Springer, Isbn: 9783540378815.
- LIU, B. 2010. Sentiment analysis and subjectivity. In: INDURKHYA, N. and DAMERAU, F. (eds.) *Handbook of Natural Language Processing*. Boca Raton: CRC Press, Isbn: 9781420085921
- LIU, B. 2012a. Opinion Search and Retrieval. *Sentiment Analysis and Opinion Mining*. Morgan & Claypool, Isbn: 9781608458844.
- LIU, B. 2012b. *Sentiment Analysis and Opinion Mining*, Morgan & Claypool, Isbn: 9781608458844.
- LIU, B. 2015. *Sentiment analysis: Mining opinions, sentiments, and emotions*, Cambridge University Press, Isbn: 1316298329.
- LIU, S., ZHU, W., XU, N., LI, F., CHENG, X.-Q., LIU, Y. and WANG, Y. 2013. Proceedings of the 22nd international conference on World Wide Web companion. International World Wide Web Conferences Steering Committee, pp: 105-106.
- MAAS, A. L., DALY, R. E., PHAM, P. T., HUANG, D., NG, A. Y. and POTTS, C. 2011. Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1. Association for Computational Linguistics, pp: 142-150.
- MANNING, C. D., RAGHAVAN, P. and SCHÜTZE, H. 2008a. Scoring, term weighting and the vector space model. *Introduction to information retrieval*. Cambridge university press Cambridge, Isbn: 9781139472104.
- MANNING, C. D., RAGHAVAN, P. and SCHÜTZE, H. 2008b. Support vector machines and machine learning on documents. *Introduction to Information Retrieval*. Cambridge University Press, Isbn: 9781139472104.
- MARTIN-VALDIVIA, M.-T., MARTINEZ-CAMARA, E., PEREA-ORTEGA, J.-M. and URENA-LOPEZ, L. A. 2013. Sentiment polarity detection in Spanish reviews combining supervised and unsupervised approaches. *Expert Systems with Applications*, Vol. 40, pp. 3934-3942.
- MATLAB. 1994. *The MathWorks, Inc.*.
- MCCALLUM, A. and NIGAM, K. 1998a. The International Association for the Advancement of Artificial Intelligence (AAAI) : workshop on learning for text categorization. Citeseer, pp: 41-48.
- MCCALLUM, A. and NIGAM, K. 1998b. IN AAAI-98 WORKSHOP ON LEARNING FOR TEXT CATEGORIZATION. pp: 41-48.
- MCDONALD, R., CRAMMER, K. and PEREIRA, F. 2005. Proceedings of the 43rd annual meeting on association for computational linguistics. Association for Computational Linguistics, pp: 91-98.
- MCDONALD, R., HANNAN, K., NEYLON, T., WELLS, M. and REYNAR, J. 2007. 45th Annual Meeting of the Association of Computational Linguistics, Prague, Czech Republic. Association for Computational Linguistics, pp: 432-439.
- MEENA, A. and PRABHAKAR, T. V. 2007. Sentence Level Sentiment Analysis in the Presence of Conjuncts Using Linguistic Analysis. In: AMATI, G., CARPINETO, C. and ROMANO, G. (eds.) *Advances in Information Retrieval*. Springer Berlin Heidelberg, Isbn: 9783540714941.
- MELVILLE, P., GRYC, W. and LAWRENCE, R. D. 2009. Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining, Paris, France. 1557156: ACM, pp: 1275-1284.



- 
- MERWE, L. V. D. and VILJOEN, C. 2000. Descriptive Statistic: Organising and Describing Data. *Elementary Statistics: Vol 2*. Pearson Education South Africa, Isbn: 9781868910755.
- MIAO, Y.-Q. and KAMEL, M. 2011. Pairwise optimized Rocchio algorithm for text categorization. *Pattern Recognition Letters*, Vol. 32, pp. 375-382.
- MIHALCEA, R. and LIU, H. 2006. Proceedings of The International Association for the Advancement of Artificial Intelligence (AAAI) Spring Symposium on Computational Approaches to Weblogs. pp.
- MILLER, G. A. 1995. WordNet: a lexical database for English. *Communications of the ACM*, Vol. 38, pp. 39-41.
- MILLER, S. 2005. Data presentation adn the normal distribution. *Experimental Design and Statistics*. Taylor & Francis, Isbn: 9781134954629.
- MIRSKY, M. J. 2011. *The Drama in Shakespeare's Sonnets: 'A Satire to Decay'*, Fairleigh Dickinson University Press, Isbn: 9781611470277.
- MISHNE, G. 2005. Proceedings of ACM Special Interest Group on Information Retrieval (SIGIR) 2005 Workshop on Stylistic Analysis of Text for Information Access. pp.
- MISHNE, G. and GLANCE, N. 2006. The International Association for the Advancement of Artificial Intelligence (AAAI) 2006 Spring Symposium on Computational Approaches to Analysing Weblogs, Palo Alto, California, USA. pp: 301-304.
- MONTGOMERY, D. C. 2013a. Experiments with Blocking Factors. *Design and Analysis of Experiments*. Wiley, Isbn: 9781118097939.
- MONTGOMERY, D. C. 2013b. Factorial Experiments. *Design and Analysis of Experiments*. Wiley, Isbn: 9781118097939.
- MONTGOMERY, D. C. and RUNGER, G. C. 2007. Design and Analysis of Single-Factor Experiments: The Analysis of Variance. *Applied statistics and probability for engineers*. Wiley, Isbn: 9780471745891.
- MORAES, R., VALIATI, J. F. and NETO, W. P. G. 2013. Document-level sentiment classification: An empirical comparison between SVM and ANN. *Expert Systems with Applications*, Vol. 40, pp. 621-633.
- MOYER, M. W. 2014. *Twitter to Release All Tweets to Scientists* [Online]. Available: <http://www.scientificamerican.com/article/twitter-to-release-all-tweets-to-scientists-a-trove-of-billions-of-tweets-will-be-a-research-boon-and-an-ethical-dilemma/> [Accessed 22 March].
- MUHAMMAD, A., WIRATUNGA, N., LOTHIAN, R. and GLASSEY, R. 2013. Workshop co-located with AI-2013 Thirty-third SGAI International Conference on Artificial Intelligence (BCS SGAI), Cambridge, UK. pp: 7-18.
- NADEAU, D., SABOURIN, C., KONINCK, J. D., MATWIN, S. and TURNEY, P. D. 2006. In: Proceedings of the Workshop on Computational Aesthetics at the Twenty-First National Conference on Artificial Intelligence, Boston, Massachussets, USA. pp.
- NATURAL LANGUAGE PROCESSING AND INFORMATION RETRIEVAL GROUP AT UNED. 2010. *WePS: searching information about entities in the web* [Online]. Available: <http://nlp.uned.es/weps/>.
- NG, A. Y. 1997. ICML. pp: 245-253.
- NIELSEN, F. A. 2011a. 7th International Conference Mechatronic Systems and Materials (MSM 2011), Kaunas, Lithuania. pp.
- NIELSEN, F. Å. 2011b. A new ANEW: Evaluation of a word list for sentiment analysis in microblogs. *arXiv preprint arXiv:1103.2903*, Vol.
- NIGAM, K. and HURST, M. 2004. AAAI spring symposium on exploring attitude and affect in text. pp: 598-603.
-

- 
- NIGAM, K., LAFFERTY, J. and MCCALLUM, A. 1999. IJCAI-99 workshop on machine learning for information filtering. pp: 61-67.
- NORTON, P. C., SAMUEL, A., AITEL, D., FOSTER-JOHNSON, E., RICHARDSON, L., DIAMOND, J., PARKER, A. and ROBERTS, M. 2005. *Beginning Python*, Wiley, Isbn: 9780471760313.
- OHANA, B. and TIERNEY, B. 2009. Annual Information Technology & Telecommunications Conference (IT&T), Dublin, Ireland. pp.
- OLIPHANT, T. E. 2006. *A Guide to NumPy*, Trelgol Publishing USA, Isbn: -.
- ORTIZ, C. E. 2010. Introduction to Facebook APIs. <http://www.ibm.com/>.
- OSBORNE, M. 2002. Proceedings of the ACL-02 Workshop on Automatic Summarization-Volume 4. Association for Computational Linguistics, pp: 1-8.
- OXFORD ENGLISH DICTIONARY. 2015a. *Poem* [Online]. Available: <http://www.oxforddictionaries.com/definition/english/poem> [Accessed 10 June 2015].
- OXFORD ENGLISH DICTIONARY. 2015b. *Sonnet* [Online]. Available: <http://www.oxforddictionaries.com/definition/english/sonnet> [Accessed 10 June 2015].
- PAGE, L., BRIN, S., MOTWANI, R. and WINOGRAD, T. 1998. Proceedings of the 7th International World Wide Web Conference, Brisbane, Australia. Pageetal98, pp: 161-172.
- PAK, A., BERNHARD, D., PAROUBEK, P. and GROUIN, C. 2012. A combined Approach to emotion Detection in suicide notes. *Biomedical informatics insights*, Vol. 5, pp. 105.
- PANG, B. and LEE, L. 2004. Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics, Barcelona, Spain. 1218990: Association for Computational Linguistics, pp: 271.
- PANG, B. and LEE, L. 2008. Opinion Mining and Sentiment Analysis. *Foundations and Trends in Information Retrieval*, Vol. 2, pp. 1-135.
- PANG, B., LEE, L. and VAITHYANATHAN, S. 2002. Proceedings of the ACL-02 conference on Empirical methods in natural language processing - Volume 10. 1118704: Association for Computational Linguistics, pp: 79-86.
- PATIL, B. M., JOSHI, R. C. and TOSHNIWAL, D. 2010. Impact of k-means on the performance of classifiers for labeled data. *Contemporary Computing*. Springer, Isbn: 9783642148330.
- PAUL. 2015. *London at a Slower Pace* [Online]. <http://www.tripadvisor.co.uk/>. Available: [http://www.tripadvisor.co.uk/ShowUserReviews-g186338-d553603-r278552569-The\\_London\\_Eye-London\\_England.html#CHECK\\_RATES\\_CONT](http://www.tripadvisor.co.uk/ShowUserReviews-g186338-d553603-r278552569-The_London_Eye-London_England.html#CHECK_RATES_CONT).
- PECK, R., OLSEN, C. and DEVORE, J. 2001. Hypothesis Testing Using a Single Sample. *Introduction to statistics and data analysis*. Pacific Grove, CA: Duxbury, Isbn: 9780534370923.
- PEDREGOSA, F., VAROQUAUX, G., GRAMFORT, A., MICHEL, V., THIRION, B., GRISEL, O., BLONDEL, M., PRETTENHOFER, P., WEISS, R., DUBOURG, V., VANDERPLAS, J., PASSOS, A., COURNAPEAU, D., BRUCHER, M., PERROT, M. and DUCHESNAY, E. 2011. Scikit-learn: Machine Learning in Python. *Machine Learning Research*, Vol. 12, pp. 2825-2830.
- PENNEBAKER, J. W., FRANCIS, M. E. and BOOTH, R. J. 2001. Linguistic inquiry and word count: LIWC 2001. *Mahway: Lawrence Erlbaum Associates*, Vol. 71, pp. 2001.
- PESTIAN, J. P., MATYKIEWICZ, P., LINN-GUST, M., SOUTH, B., UZUNER, O., WIEBE, J., COHEN, K. B., HURDLE, J. and BREW, C. 2012. Sentiment Analysis of Suicide Notes: A Shared Task. *Biomed Inform Insights*, Vol. 5, pp. 3-16.
-

- PETZ, G., KARPOWICZ, M., FURSCHUB, H., AUINGER, A., WINKLER, S. M., SCHALLER, S. and HOLZINGER, A. 2012. On Text Preprocessing for Opinion Mining Outside of Laboratory Environments. *In: HUANG, R., GHORBANI, A., PASI, G., YAMAGUCHI, T., YEN, N. and JIN, B. (eds.) Active Media Technology*. Springer Berlin Heidelberg, Isbn: 9783642352355.
- PFITZNER, R., GARAS, A. and SCHWEITZER, F. 2012. The 6th The International Association for the Advancement of Artificial Intelligence (AAAI) Conference on Weblogs and Social Media, Dublin, Ireland. pp.
- POLAT, K. and GUNES, S. 2009. A novel hybrid intelligent method based on C4. 5 decision tree classifier and one-against-all approach for multi-class classification problems. *Expert Systems with Applications*, Vol. 36, pp. 1587-1592.
- POLIKAR, R. 2012. Ensemble learning. *Ensemble Machine Learning*. Springer, Isbn: 1441993258.
- PONOMAREVA, N. 2014. *Graph-based approaches for semi-supervised and cross-domain sentiment analysis*. Doctor of Philosophy, University of Wolverhampton.
- POOLE, D. L. and MACKWORTH, A. K. 2010. Cross Validation. *Artificial Intelligence: Foundations of Computational Agents*. Cambridge University Press, Isbn: 9780521519007.
- POPESCU, A.-M. and ETZIONI, O. 2007. Extracting product features and opinions from reviews. *Natural language processing and text mining*. Springer, Isbn: 184628175X.
- POWERS, D. M. 2011. Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation. *Machine Learning RTechnologies*, Vol. 2, pp. 37-63.
- PREETHI, T., DEVI, K. N. and BHASKARAN, V. M. 2012. International Conference on Recent Trends In Information Technology (ICRTIT 2012). pp: 497-501.
- PRINCETON UNIVERSITY. 2010. *WordNet*. Available: <http://wordnet.princeton.edu/>
- PRODROMIDIS, A., CHAN, P. and STOLFO, S. 2000. Meta-learning in distributed data mining systems: Issues and approaches. *Advances in distributed and parallel knowledge discovery*, Vol. 3.
- PYTHON SOFTWARE FOUNDATION. 2001. *Python*.
- QADIR, A. and RILOFF, E. 2013. the 4th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis (WASSA 2013). pp: 2-11.
- QUINLAN, J. R. 1986. Simplifying decision trees. *International Journal of Man-Machine Studies*, Vol. 27, pp. 221-234.
- QUINLAN, J. R. 1993. *C4.5: Programs for Machine Learning*, Morgan Kaufmann, Isbn: 9781558602380.
- R CORE TEAM. 2015. *R: A language and environment for statistical computing*. R Foundation for Statistical Computing [Online]. Vienna, Austria. Available: <http://www.R-project.org/>.
- RAEZ, A. M., CAMARA, E. M., MARTIN-VALDIVIA, M. T. and LOPEZ, L. A. U. 2012. Proceedings of the 3rd Workshop in Computational Approaches to Subjectivity and Sentiment Analysis, Jeju, Republic of Korea. Association for Computational Linguistics, pp: 3-10.
- RAO, D., YAROWSKY, D., SHREEVATS, A. and GUPTA, M. 2010. Proceedings of the 2nd international workshop on Search and mining user-generated contents. ACM, pp: 37-44.
- READ, J. 2009. *Weakly Supervised Techniques for the Analysis of Evaluation in Text*. PhD Thesis.
- REFAEE, E. and RIESER, V. 2014. In 9 th International Conference on Language Resources and Evaluation (LREC'14). pp.
- REFAEILZADEH, P., TANG, L. and LIU, H. 2009. Cross-validation. *Encyclopedia of database systems*. Springer, Isbn: 0387355448.

- 
- RICHARDSON, M., PRAKASH, A. and BRILL, E. 2006. Proceedings of the 15th international conference on World Wide Web, Edinburgh, Scotland. 1135881: ACM, pp: 707-715.
- RICHMOND, W. K. 1965. *Teachers and machines: an introduction to the theory and practice of programmed learning*, Collins, Isbn: -.
- RIJSBERGEN, C. J. V. 1979. *Information Retrieval*, Butterworth-Heinemann, Isbn: 0408709294.
- RIVERS, C. M. and LEWIS, B. L. 2014. Ethical research standards in a world of big data. *F1000Research*, Vol. 3.
- ROJAS, R. 1996. Perceptron Learning. *Neural Networks: A Systematic Introduction*. Springer Berlin Heidelberg, Isbn: 9783540605058.
- ROKACH, L. 2005. Ensemble Methods for Classifiers. In: MAIMON, O. and ROKACH, L. (eds.) *Data Mining and Knowledge Discovery Handbook*. Springer US, Isbn: 9780387244358.
- ROKACH, L. 2009. Chapter 3 Ensemble Classification. *Pattern Classification Using Ensemble Methods*. Singapore: World Scientific Publishing Company.
- ROKACH, L. 2010. Ensemble-based classifiers. *Artificial Intelligence Review*, Vol. 33, pp. 1-39.
- ROSENBLATT, F. 1957. *The perceptron, a perceiving and recognizing automaton Project Para*, Cornell Aeronautical Laboratory, Isbn: 03604012.
- ROWSTRON, A., NARAYANAN, D., DONNELLY, A., O'SHEA, G. and DOUGLAS, A. 2012. Proceedings of the 1st International Workshop on Hot Topics in Cloud Data Processing, Bern, Switzerland. 2169092: ACM, pp: 1-5.
- ROY, B. N. 2002. Phase Equilibria and Phase Transition. *Fundamentals of Classical and Statistical Thermodynamics*. Wiley, Isbn: 9780470843161.
- RUMSEY, D. J. 2007. Pointing Out Correlations with Spearman's Rank. *Intermediate Statistics For Dummies*. Wiley, Isbn: 9780470147740.
- RUSHING, H., KARL, A. and WISNOWSKI, J. 2013. Simple Comparative Experiments. *Design and Analysis of Experiments by Douglas Montgomery: A Supplement for Using JMP*. SAS Institute, Isbn: 9781612908014.
- SAFAVIAN, S. R. and LANDGREBE, D. 1991. A survey of decision tree classifier methodology. *IEEE transactions on systems, man, and cybernetics*, Vol. 21, pp. 660-674.
- SAFKO, L. 2010. *The Social Media Bible: Tactics, Tools, and Strategies for Business Success*, Wiley, Isbn: 9780470912706.
- SAIF, H., HE, Y. and ALANI, H. 2012. Semantic sentiment analysis of twitter. *The Semantic Web-ISWC 2012*. Springer, Isbn: 3642351751.
- SALTON, G. 1971. *The SMART retrieval system : experiments in automatic document processing*, Prentice-Hall, Inc., Isbn: 0138145253
- SANDERSON, D. 2012. Guilty: the fanatics hellbent on carnage at Christmas. 2 February 2012. <http://www.thetimes.co.uk/>: The Times.
- SARKER, S. K. 1998. *Shakespeare's Sonnets*, Atlantic Publishers & Distributors (P) Limited, Isbn: 9788171567256.
- SCHAPIRE, R. E. 1990. The strength of weak learnability. *Machine learning*, Vol. 5, pp. 197-227.
- SCHAPIRE, R. E. and SINGER, Y. 2000. BoosTexter: A boosting-based system for text categorization. *Machine learning*, Vol. 39, pp. 135-168.
- SCHMID, H. 1995. Treetagger| a language independent part-of-speech tagger. *Institut für Maschinelle Sprachverarbeitung, Universität Stuttgart*, Vol. 43, pp. 28.
- SEEWALD, A. K. and FÜRNKRANZ, J. 2001. An evaluation of grading classifiers. *Advances in Intelligent Data Analysis*. Springer, Isbn: 9783540425816.
- SEWELL, M. 2008. Ensemble learning. *Research Note at University College London, Department of Computer Science*, Vol. 11.
-



- SHAIKH, M. A., PRENDINGER, H. and MITSURU, I. 2007. Proceedings of the 2nd international conference on Affective Computing and Intelligent Interaction, Lisbon, Portugal. 1422190: Springer-Verlag, pp: 191-202.
- SHAKESPEARE, W. 1734. *Twelfth-night: Or, what You Will. By Mr. William Shakespear*, J. Tonson, and the rest of the proprietors; and sold, Isbn: -.
- SHAKESPEARE, W. 1816. *The Works of William Shakspeare...: Collated Verbatim with the Most Authentic Copies, and Revised, with the Corrections and Illustrations of Various Commentators*, The proprietors, Isbn: -.
- SHAPIRO, S. S. and WILK, M. B. 1965. An analysis of variance test for normality (complete samples). *Biometrika*, Vol. 52, pp. 591-611.
- SIEGEL, S. and CASTELLAN, N. J. 1988. *Nonparametric Statistics for the Behavioral Sciences*, McGraw-HiU Book Company, Isbn: -.
- SING, J. K., SARKAR, S. and MITRA, T. K. 2012. 3rd National Conference on Emerging Trends and Applications in Computer Science (NCETACS). pp: 38-40.
- SISWANTO, E. and KHODRA, M. L. 2013. International Conference on Information Technology and Electrical Engineering (ICITEE). pp: 176-180.
- SRIHARI, S. N., GOVINDARAJU, V. and SHEKHAWAT, A. 1993. Proceedings of the Second International Conference on Document Analysis and Recognition. pp: 291-294.
- ST-ONGE, M., LORTIE-LUSSIER, M., MERCIER, P., GRENIER, J. and KONINCK, J. D. 2005. Emotions in the Diary and REM Dreams of Young and Late Adulthood Women and Their Relation to Life Satisfaction. *Journal Dreaming*, Vol. 15, pp. 116-128.
- STONE, P. J., DUNPHY, D. C., SMITH, M. S. and OLGILVIE, D. M. 1968. The General Inquirer: A Computer Approach to Content Analysis. *Journal of Regional Science*, Vol. 8, pp. 113-116.
- STRAPPARAVA, C. and MIHALCEA, R. 2007. Proceedings of the 4th International Workshop on Semantic Evaluations. Association for Computational Linguistics, pp: 70-74.
- STRAPPARAVA, C. and VALITUTTI, A. 2004. The International Conference on Language Resources and Evaluation. pp: 1083-1086.
- SUN, Q. and PFAHRINGER, B. 2011. Bagging ensemble selection. *AI 2011: Advances in Artificial Intelligence*. Springer, Isbn: 9783642258312.
- SUNNI, I. and WIDYANTORO, D. H. 2012. Analisis Sentimen dan Ekstraksi Topik Penentu Sentimen pada Opini terhadap Tokoh Publik. *Jurnal Sarjana ITB bidang Teknik Elektro dan Informatika*, Vol. 1.
- TABOADA, M., BROOKE, J. and STEDE, M. 2009. The 10th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL), London, United Kingdom. 1708385: Association for Computational Linguistics, pp: 62-70.
- TAKAMURA, H., INUI, T. and OKUMURA, M. 2007. Proceedings of Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Rochester, NY. Association for Computational Linguistics, pp: 292-299.
- TAN. 2015. *Most boring movie ever* [Online]. [http://www.amazon.co.uk/product-reviews/B00006FMGR/ref=cm\\_cr\\_pr\\_hist\\_1?ie=UTF8&filterBy=addOneStar&showViewpoints=0&sortBy=bySubmissionDateDescending](http://www.amazon.co.uk/product-reviews/B00006FMGR/ref=cm_cr_pr_hist_1?ie=UTF8&filterBy=addOneStar&showViewpoints=0&sortBy=bySubmissionDateDescending). Available: [http://www.amazon.co.uk/product-reviews/B00006FMGR/ref=cm\\_cr\\_pr\\_hist\\_1?ie=UTF8&filterBy=addOneStar&showViewpoints=0&sortBy=bySubmissionDateDescending](http://www.amazon.co.uk/product-reviews/B00006FMGR/ref=cm_cr_pr_hist_1?ie=UTF8&filterBy=addOneStar&showViewpoints=0&sortBy=bySubmissionDateDescending).
- TAN, L. K., NA, J.-C., THENG, Y.-L. and CHANG, K. 2011. Workshop on Social Web Search and Mining (SWSM 2011), Beijing, China. pp.
- TAN, P. N., STEINBACH, M. and KUMAR, V. 2014. Cluster Analysis: Basic Concepts and Algorithms. *Introduction to Data Mining*. Pearson Education, Limited, Isbn: 9781292026152.

- TAN, S., CHENG, X., GHANEM, M. M., WANG, B. and XU, H. 2005. Proceedings of the 14th ACM international conference on Information and knowledge management. ACM, pp: 469-476.
- TAN, S., CHENG, X., WANG, Y. and XU, H. 2009. Adapting naive bayes to domain adaptation for sentiment analysis. *Advances in Information Retrieval*. Springer, Isbn: 9783642009570.
- TANG, B., CHEN, Q., WANG, X. and WANG, X. 2010. Reranking for Stacking Ensemble Learning. In: WONG, K., MENDIS, B. S. and BOUZERDOUM, A. (eds.) *Neural Information Processing. Theory and Algorithms*. Springer Berlin Heidelberg, Isbn: 9783642175367.
- TANG, H., TAN, S. and CHENG, X. 2009. A survey on sentiment detection of reviews. *Expert Systems with Applications*, Vol. 36, pp. 10760-10773.
- THAKKER, D., OSMAN, T. and LAKIN, P. 2009. GATE JAPE Grammar Tutorial Version 1.0.
- THE STANFORD NATURAL LANGUAGE PROCESSING GROUP. 2002. *lex-parser*. Available: <http://nlp.stanford.edu/software/lex-parser.shtml>
- THELWALL, M., BUCKLEY, K., PALTOGLOU, G. and CAI, D. 2010a. Sentiment strength detection in short informal text. *Journal of the American Society for Information Science and Technology*, Vol. 61, pp. 2544-2558.
- THELWALL, M., BUCKLEY, K., PALTOGLOU, G., CAI, D. and KAPPAS, A. 2010b. *SentiStrength*. Available: <http://sentistrength.wlv.ac.uk/>
- THET, T. T., NA, J.-C. and KHOO, C. S. 2010. Aspect-based sentiment analysis of movie reviews on discussion boards. *Journal of Information Science*, Vol. 36, pp. 1-26.
- THET, T. T., NA, J.-C., KHOO, C. S. and SHAKTHIKUMAR, S. 2009. Proceedings of the 1st International Conference on Information and Knowledge Management (CIKM) workshop on Topic-sentiment analysis for mass opinion. ACM, pp: 81-84.
- TRENT, T. 2014. *Scientists propose tactics for ethical use of Twitter data in research studies* [Online]. Available: <http://www.vtnews.vt.edu/articles/2014/06/061014-vbi-twitterethics.html> [Accessed 22 March].
- TROUSSAS, C., VIRVOU, M., JUNSHEAN ESPINOSA, K., LLAGUNO, K. and CARO, J. 2013. 4th International Conference on Information, Intelligence, Systems and Applications (IISA). pp: 1-6.
- TSURUOKA, Y., TSUJII, J. I. and ANANIADOU, S. 2009. Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing (AFNLP). Association for Computational Linguistics, pp: 477-485.
- TUMASJAN, A., SPRENGER, T. O., SANDNER, P. G. and WELPE, I. M. 2010. Proceedings of the 4th the Association for the Advancement of Artificial Intelligence (AAAI) Conference on Weblogs and Social Media. pp: 178-185.
- TURNEY, P. D. 2001. Proceedings of the 12th European Conference on Machine Learning. 650004: Springer-Verlag, pp: 491-502.
- TURNEY, P. D. 2002. Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, Philadelphia, Pennsylvania. 1073153: Association for Computational Linguistics, pp: 417-424.
- TURNEY, P. D. and LITTMAN, M. L. 2003. Measuring praise and criticism: Inference of semantic orientation from association. *ACM Transactions on Information Systems (TOIS)*, Vol. 21, pp. 315-346.
- TWITTER, I. 2016. *Twitter Privacy Policy* [Online]. Available: <https://twitter.com/privacy?lang=en> [Accessed 22 March].

- WAGNER, J., LINGENFELSER, F., BEE, N. and ANDRÉ, E. 2011. Social signal interpretation (ssi). *KI-Kuenstliche Intelligenz*, Vol. 25, pp. 251-256.
- WALTINGER, U. 2010. Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC). pp: 1638-1642.
- WAN, X. 2008. Proceedings of the Conference on Empirical Methods in Natural Language Processing, Honolulu, Hawaii. 1613783: Association for Computational Linguistics, pp: 553-561.
- WAN, X. 2009. Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing (AFNLP), Suntec, Singapore. 1687913: Association for Computational Linguistics, pp: 235-243.
- WANG, L. and CARDIE, C. 2014. The 52nd Annual Meeting of the Association for Computational Linguistics Baltimore, USA. pp.
- WANG, W., ZHAO, Y., QIU, L. and ZHU, Y. 2014. Effects of Emoticons on the Acceptance of Negative Feedback in Computer-Mediated Communication. *Journal of the Association for Information Systems*, Vol. 15, pp. 454-483.
- WHISSELL, C. 2009. Using the Revised Dictionary of Affect in Language to quantify the emotional undertones of samples of natural language. *Psychological Reports*, Vol. 105, pp. 509-521.
- WHITE, H. 1989. Learning in artificial neural networks: A statistical perspective. *Neural computation*, Vol. 1, pp. 425-464.
- WHITEHEAD, M. and YAEGER, L. 2009. World Congress on Computer Science and Information Engineering (WRI). pp: 472-476.
- WIEBE, J., WILSON, T. and CARDIE, C. 2005. Annotating Expressions of Opinions and Emotions in Language. *Language Resources and Evaluation*, Vol. 39, pp. 165-210.
- WILCOXON, F. 1945. Individual comparisons by ranking methods. *Biometrics bulletin*, Vol. 1, pp. 80-83.
- WILLMOTT, C. J. and MATSUURA, K. 2005. Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance. *Climate research*, Vol. 30, pp. 79.
- WILSON, T., HOFFMANN, P., SOMASUNDARAN, S., KESSLER, J., WIEBE, J., CHOI, Y., CARDIE, C., RILOFF, E. and PATWARDHAN, S. 2005a. Proceedings of HLT/EMNLP on Interactive Demonstrations, Vancouver, British Columbia, Canada. 1225751: Association for Computational Linguistics, pp: 34-35.
- WILSON, T., KOZAREVA, Z., NAKOV, P., RITTER, A., ROSENTHAL, S. and STOYANOV, V. 2013. Proceedings of the 7th International Workshop on Semantic Evaluation (SemEval). Association for Computational Linguistics, pp: 312-320.
- WILSON, T., WIEBE, J. and HOFFMANN, P. 2005b. Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing, Vancouver, British Columbia, Canada. 1220619: Association for Computational Linguistics, pp: 347-354.
- WILSON, T., WIEBE, J. and HOFFMANN, P. 2009. Recognizing contextual polarity: An exploration of features for phrase-level sentiment analysis. *Computational linguistics*, Vol. 35, pp. 399-433.
- WITMER, D. F. and KATZMAN, S. L. 1997. Smile when you say that: graphic accents as gender markers in computer-mediated communication. In: FAY, S., MARGARET, M. and SHEIZAF, R. (eds.) *Network and Netplay*. MIT Press, Isbn: 9780262692069.
- WITTEN, I. H. and FRANK, E. 2005. *Data Mining: Practical Machine Learning Tools and Techniques, Second Edition*, Elsevier Science, Isbn: 9780080477022.

- WITTEN, I. H., FRANK, E. and HALL, M. A. 2011. PART III THE WEKA DATA MINING WORKBENCH. *Data Mining: Practical Machine Learning Tools and Techniques: Practical Machine Learning Tools and Techniques*. Elsevier Science, Isbn: 9780080890364.
- WOLPERT, D. H. 1992. Stacked generalization. *Neural networks*, Vol. 5, pp. 241-259.
- WOODS, G. 2013. *Wiley AP English Literature and Composition*, Wiley, Isbn: 9781118490235.
- WU, F.-Y. 1982. The potts model. *Reviews of modern physics*, Vol. 54, pp. 235.
- WU, Q., TAN, S. and CHENG, X. 2009. Proceedings of the ACL-IJCNLP 2009 Conference Short Papers, Suntec, Singapore. 1667681: Association for Computational Linguistics, pp: 317-320.
- YERVA, S. R., MIKLOS, Z. and ABERER, K. 2010. Third Web People Search Evaluation Forum (WePS-3). pp.
- YONG, S. L., HAGENBUCHNER, M. and TSOI, A. C. 2008. Proceedings of the 2008 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology - Volume 03. 1487317: IEEE Computer Society, pp: 677-680.
- YU, H. and HATZIVASSILOGLU, V. 2003. Proceedings of the 2003 conference on Empirical methods in natural language processing. 1119372: Association for Computational Linguistics, pp: 129-136.
- YUAN, B., LIU, Y., LI, H., PHAN, T. T. T., KAUSAR, G., SING-BIK, C. N., SINGH, R. G., ALCOY, J. C. O., SAWHNEY, U. and HIRADHAR, P. 2013. Sentiment Classification in Chinese Microblogs: Lexicon-based and Learning-based Approaches. *International Proceedings of Economics Development and Research (IPEDR)* Vol. 68.
- ZAGIBALOV, T. 2010. *Unsupervised and knowledge-poor approaches to sentiment analysis*. University of Sussex.
- ZAPPAVIGNA, M. 2012. *Discourse of Twitter and Social Media: How We Use Language to Create Affiliation on the Web*, Bloomsbury Publishing, Isbn: 9781441123039.
- ZHU, X., GOLDBERG, A. B., BRACHMAN, R. and DIETTERICH, T. 2009a. *Introduction to Semi-Supervised Learning*, Morgan and Claypool Publishers, Isbn: 9781598295474.
- ZHU, X., GOLDBERG, A. B., BRACHMAN, R. and DIETTERICH, T. 2009b. *Introduction to Semi-Supervised Learning*, Morgan and Claypool Publishers, Isbn.



## Appendices

**Appendix I: Examples of sentiment that were expressed in the writing of poems, sonnets, histories, books and media**

The following quote is taken from the poem, 'Death Be Not Proud' by John Donne (1572-1631). This poem expresses resistance against fate and death.

“Death be not proud, though some have called thee  
Mighty and dreadful, for, thou art not so,  
For, those, whom thou think'st, thou dost overthrow,  
Die not, poore death, nor yet canst thou kill me.”

(Donne and Alford, 1839; Donne, 2013; Woods, 2013)

Poems and sonnets are simultaneously quite different and similar. A poem is a piece of writing that expresses feelings and ideas that are given intense attention through diction, rhythm and imagery (Oxford English Dictionary, 2015a). Meanwhile, a sonnet is a 14-line poem containing any of a number of formal rhyme schemes; typically containing 10 syllables per line (Oxford English Dictionary, 2015b).

The following quote is taken from Sonnet 116 from Shakespeare (1564-1616); Mirsky (2011) stated that, in Sonnet 116, the poet will pledge against all Shakespeare reservations about love and its blind folly, for its reality as a substance that can defy death.

“Let me not to the marriage of true minds  
Admit impediments. Love is not love  
Which alters when it alteration finds,  
Or bends with the remover to remove:  
O no! It is an ever-fixèd mark  
That looks on tempests and is never shaken;  
It is the star to every wandering bark,  
Whose worth's unknown, although his height be taken.  
Love's not Time's fool, though rosy lips and cheeks  
Within his bending sickle's compass come;  
Love alters not with his brief hours and weeks,  
But bears it out even to the edge of doom.  
If this be error and upon me proved,  
I never writ, nor no man ever loved.”

(Shakespeare, 1816; Sarker, 1998; Mirsky, 2011).

In histories, for example, in Christian history, there is both a prehistory of emotion in human evaluation and a history that includes biblical and secular pre-Christian sources. The history of emotion within Christianity is part of the history of emotion's role in religion in general, as found in the Bible and in classical Greece and Rome before Christ. Compared with the evaluation of emotion, Christianity's story is much shorter, being solely a human experience, with evolutionary roots in a religions sentiment going back at least to the ancient narrative, the cosmogonies, the myths of gods and demons, the sagas of birth and death, time and eternity, good and evil, light and darkness, love and hate. Evidence of more recent religious practice can be found in traces in ancient texts and documents in the immediate source of Christianity: the Bible. (Corrigan, 2008)

Furthermore, another form of writing that expresses emotions is in the book; for example, the Shakespeare play, *Twelfth Night*. This text gathers into itself all that is most fragrant in the romantic comedies and the fullness of its perfection can only be discovered by examining the whole action, its characters and the neat arrangement of its situation; whereby the expression of sentiment is in the idiom of the sonnet (Evans, 2013).

The following quote is taken from *Twelfth Night*, as Viola describes the beauty of Olivia:

Lady, you are the cruell'st she alive,  
If you will lead these graces to the grave  
And leave the world no copy"

(Shakespeare, 1734; Evans, 2013)

Furthermore, emotion can be expressed in media such as in news, news headlines and customer reviews. The news titles are often written to provoke reader emotions (Strapparava and Mihalcea, 2007). The following samples of news headlines:

"Growing Unarmed Battalion in Qaeda Army Is Using Internet to Get the Message Out"

(Fattah, 2006)

“Home office fails to shut down a single extremist website in two years.”

(Hope, 2009)

Besides the headlines, the inside news story is also written using emotional expression. The following quote is taken from The Times:

“Two of the nine-strong network of young Muslim men were captured conducting surveillance on London targets including the London Eye, Big Ben and the Church of Scientology. They also had advanced plans to plant a bomb in London Stock Exchange lavatories.”

(Sanderson, 2012)

Furthermore, emotions are expressed commonly in customer reviews on sites such as Amazon and Tripadvisor. Below are customer reviews of ‘the London Eye’ taken from Tripadvisor:

“Admittedly it was a bank holiday weekend but a four hour queue with the kids is a lot of sightseeing time wasted so we decided not to ride the wheel. It was good to see and we took many photos but the nearby Westminster Bridge, Big Ben and the Houses of Parliament were great photo opportunities also. If you are able to book in advance then that's what I would recommend!!!”

(Jose, 2015)

“Booked online and took the priority queuing option. This means you have to be there just 15 minutes before your booked time. Well worth the extra cost. The surroundings can very congest. But considering how busy this attraction is the system works brilliantly. Good organization with refreshments and toilets available. A camera is essential. The half hour ride may not seem long, but I assure you the pace is great.”

(Paul, 2015)

The following quotes are examples of customer reviews of the film ‘8 Mile [DVD]’ taken from Amazon:

“Wasn't sure what to expect from this but it was a really nice surprise. Eminem plays down and out rapper Rabbit, trying to find some direction with no money and little prospects, in his first acting role. The film is gritty, showing an area with little hope for people to make anything of themselves, a trailer park culture that feels very believable. Rap battles are a mark of respect and Rabbit has to prove himself against a hostile audience which leads to a great ending where Eminem excels into his comfort zone. 5 stars, thoroughly enjoyed.”

(Bear, 2015)

“Marshall is a very self-centered guy, always rapping about himself how bad his life is even though he's a millionaire, how bad his childhood was even though he was not the only one with a bad one among other stuff. But now he felt it necessary to make a movie about his life and star in it himself so we can see all the ups and downs of his rise to fame and boy did that suck, such a boring movie and the general point of this I would of thought would be to inspire people who want to become rappers and the movie just doesn't inspire in the slightest. So I really don't see what the point of this movie was?”

(Tan, 2015)

## Appendix II: List of Stanford Part-Of-Speech Tagging

At the following, there are tags and descriptions of part-of-speech (Chen and Lonardi, 2009) :

Tag	Description
NN	Noun singular or mass
NNS	Noun plural
NNP	Proper noun singular
NNPS	Proper noun plural
JJ	Adjective
JJR	Adjective comparative
JJS	Adjective superlative
CC	Coordinating conjunction
IN	Preposition or subordinating conjunction
TO	To
CD	Cardinal number
DT	Determiner
EX	Existential there
FW	Foreign word
LS	List item marker
NP	Noun phrase
VP	Verb phrase
MD	Modal
PDT	Predeterminer
POS	Possessive ending
PRP	Personal pronoun
PRP	Possessive pronoun
RB	Adverb
RBR	Adverb comparative
RBS	Adverb superlative
RP	Particle
SYM	Symbol
VB	Verb base form
VBD	Verb past tense
VBG	Verb gerund or present participle

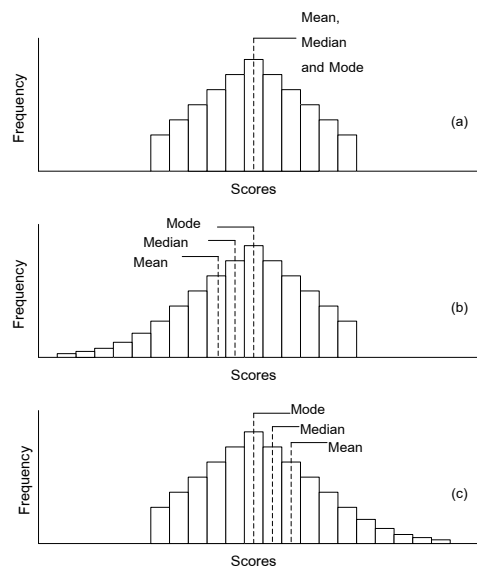
## Appendix III: Simple Comparative Experimental Design

### Simple Comparative Experimental Design

The simple comparative uses in the experiment that considers a comparison of a single factor with two levels of factors or called two treatments are provided. The designs used most commonly are basic statistic concepts and hypothesis testing (Rushing *et al.*, 2013).

#### 1. Basic Statistic Concepts

Two basic measurements in the basic statistic are mean, median and mode. The mean is the numerical average of the group score. Mean can be calculated by summarising the score and dividing it by the number of scores. When the group scores have been arranged from lowest to highest, medium is the middle score that can divide the scores into two equal parts. The mode is the most common scores in the group. Their relationship can be explained by using a histogram, as illustrated in Figure 24. The symmetric curve (a) means that the values of mean, median and mode are the same and they lie in the centre of distribution. When the curve is skewed to the right (c), it means the value of the mean is the lowest while the value of mode is the highest. In contrast, if the value of mode is the lowest and the value of the mean is the highest, the curve will skew to the left (b).



**Figure A1:**The relationship between mean, median and mode  
(Merwe and Viljoen, 2000; Miller, 2005)

## 2. Hypothesis Testing

The statistic testing used to assist and support the claim is called hypothesis testing. Before testing the hypothesis, the null hypothesis will specify a particular hypothesised value of  $\mu$ , which is initially assumed to be true (Gosling, 1995; Peck *et al.*, 2001).

$$H_0: \mu = \text{hypothesised value}$$

The alternative hypothesis will have one of the following forms, depending on the research question being addressed.

$$H_0: \mu > \text{hypothesised value}$$

$$H_0: \mu < \text{hypothesised value}$$

$$H_0: \mu \neq \text{hypothesised value}$$

If the sample size  $n$  is equal or lower than 30, t-test will be used.

$$t = \frac{\bar{X} - \mu}{s/\sqrt{n}} \quad (1)$$

Source: (Gosling, 1995; Peck *et al.*, 2001)

Conversely, if the sample size  $n$  is more than 30, the z-test will be used.

$$z = \frac{\bar{X} - \mu}{s/\sqrt{n}} \quad (2)$$

Source: (Gosling, 1995; Peck *et al.*, 2001)

where  $\bar{X}$  and  $s$  are the value of sample mean and sample standard deviation from the random sample.



## Appendix IV: Balance Incomplete Block Design

Balanced incomplete block design (BIBD) is an incomplete block design, whereby not all treatments are present in every block (Chow and Liu, 2004), as shown in Table 20.

Treatments (Methods)	Block			
	1	2	3	4
1	$y_{11}$	$y_{12}$	—	$y_{14}$
2	—	$y_{22}$	$y_{23}$	$y_{24}$
3	$y_{31}$	—	$y_{33}$	$y_{34}$

**Table 1:** Data records of BIBD

For analysis of variance (ANOVA) for BIBD, the forms presented in Table 21 are used (Chatfield, 1983a; Montgomery, 2013a).

Source of variation	Sum of squares	Degrees of freedom	Mean square	$F_0$
<b>Treatments (Methods)</b>	$SS_{Treatment(adjusted)}$	$a - 1$	$\frac{SS_{Treatments(adjusted)}}{a - 1}$	$\frac{MS_{Treatments(adjusted)}}{MS_E}$
<b>Blocks</b>	$SS_{Blocks}$	$b - 1$	$\frac{SS_{Blocks}}{b - 1}$	
<b>Error</b>	$SS_{E(adjusted)}$	$N - a - b + 1$	$\frac{SS_E}{N - a - b + 1}$	
<b>Total</b>	$SS_T$	$N - 1$		

**Table 2:** ANOVA for BIBD  
(Chatfield, 1983a; Montgomery, 2013a)

The formulae for the sum of squares in ANOVA for RCBD are:

$$SS_{Treatment(adjusted)} = \frac{k \sum_{i=1}^a Q_i^2}{\lambda a} \quad (5)$$

$$Q_i = y_{i.} - \frac{1}{k} \sum_{j=1}^b n_{ij} y_{.j} \quad (6)$$

$$SS_{Blocks} = \frac{1}{k} \sum_{j=1}^b y_{.j}^2 - \frac{y_{...}^2}{ab} \quad (3)$$

$$SS_E = SS_T - SS_{Treatments(adjusted)} - SS_{Blocks} \quad (4)$$

$$SS_T = \sum_{i=1}^a \sum_{j=1}^b y_{ij}^2 - \frac{y_{...}^2}{ab} \quad (5)$$

Source: (Chatfield, 1983a; Montgomery, 2013a)

For computing the degrees of freedom in ANOVA for BIBD,  $a$  refers to the number of treatments, while  $b$  refers to the number of blocks. In addition, it is assumed that each block contains treatments  $k$ .  $y_{ij}$  is the  $i$ th observation in the  $j$ th block.  $Q_i$  is the adjusted total for the  $i$  treatment with  $n_{ij} = 1$  if treatment  $i$  appear in block  $j$  and  $n_{ij} = 0$  otherwise.  $y_{i.}$  is a total of all observations taken under method  $i$ ,  $y_{.j}$  is the total in the  $j$ th block.  $y_{...}$  is a grand total of all observations.  $N$  is the total number of observations which equal  $ab$ ,  $F_0$  is used for testing the null hypothesis that the treatment effects are all zero.

## **Appendix V: Ethical Issue and Approval Confirmation**

Ethical approval for this study was granted by Northumbria University on 13 April 2012.

As public Tweets were used, Twitter Inc. was asked (Appendix VI) about ethical considerations. Twitter Inc. replied, if the users make their profile visible to the public when posting and displaying any content, they are allowing the public to use, copy and modify their content in anyway, including third parties such as Google, search engines, Twitter API can view, save or copy those contents that public via Tweets.

### **Ethic Approval**

**From:** Research Ethics Submission Portal [web.form@northumbria.ac.uk]  
**Sent:** 13 April 2012 09:06  
**To:** tawunrat.chalothorn  
**Subject:** Research Project - Approval Granted

## **Research Project - Approval Granted**

This is an automated email from the Research Ethics Submission Tool. Research project **RE02-02-12658** has been granted internal ethical approval.

## Appendix VI: Twitter Ethical Issues

In the Twitter's Privacy Policy which effect from 27<sup>th</sup> January 2016 (Twitter, 2016) states that,

**“Tweets, Following, Lists and other Public Information:** Our Services are primarily designed to help you share information with the world. *Most of the information you provide us through the Twitter Services is information you are asking us to make public.* Your public information includes the messages you Tweet; the metadata provided with Tweets, such as when you Tweeted and the client application you used to Tweet; the language and time zone associated with your account; and the lists you create, people you follow, Tweets you mark as likes or Retweet, and many other bits of information that result from your use of the Twitter Services. We may use this information to make inferences, like what topics you may be interested in, and to customize the content we show you, including ads. Our default is almost always to make the information you provide through the Twitter Services public for as long as you do not delete it, but we generally give you settings or features, like direct messages, to make the information more private if you want. You can change the language and time zone associated with your account at any time using your account settings. *The Twitter Services broadly and instantly disseminate your public information to a wide range of users, customers, and services. For instance, your public user profile information and public Tweets are immediately delivered via SMS and our APIs to our partners and other third parties, including search engines, developers, and publishers that integrate Twitter content into their services, and institutions such as universities and public health agencies that analyse the information for trends and insights.* When you share information or content like photos, videos, and links via the Services, you should think carefully about what you are making public.”

(Twitter, 2016)

## #17083677 Twitter Support: update on "Ethical issues"

YLee <notifications-support@twitter.zendesk.com>

Fri 07/03/2014 21:36

^\_^ keep it \*\*

To: tawunrat.chalothorn <tawunrat.chalothorn@northumbria.ac.uk>;

##- Please type your reply above this line -##

---

**YLee, Mar 07 01:36 PM:**

Hello,

As indicated when you create a new account, your Twitter profile is public by default. Unless your account is changed to protected from your account settings, your Tweets are publicly visible on your profile page, in Twitter search, and through the Twitter API. As soon as they have been made publicly available, third parties (such as Google and other search engines) can find, view, and even save or archive these publicly visible Tweets—like other information on the internet.

If you want your Tweets to only be available to approved followers, you can set your account to protected. Tweets posted by a protected account are only visible to approved followers and not otherwise publicly available to third parties. This help page has more information about public and protected accounts: <http://support.twitter.com/articles/14016>.

Thanks,

YLee  
Twitter Trust & Safety

---

**tawunrat.chalothorn, Mar 07 05:31 AM:**

Hello

?

I am a PhD researcher at University of Northumbria at Newcastle Upon Tyne, UK.

I wonder that are there any ethical issues when use Twitter API to collect random tweets?

Regards  
Tawunrat Chalothorn?

---

**Appendix VII: Sample of Data Entry of RCBD ANOVA in SPSS**

	Treatments	Factor	Factors	Fscore
1	1	NB	1	81.06
2	1	SVM	2	82.62
3	1	MaxEnt	3	59.93
4	2	NB	1	38.48
5	2	SVM	2	55.77
6	2	MaxEnt	3	42.94
7	3	NB	1	34.27
8	3	SVM	2	57.15
9	3	MaxEnt	3	32.18
10	4	NB	1	33.62
11	4	SVM	2	69.56
12	4	MaxEnt	3	31.85
13	5	NB	1	62.79
14	5	SVM	2	59.70
15	5	MaxEnt	3	33.30
16	6	NB	1	29.46
17	6	SVM	2	37.64
18	6	MaxEnt	3	30.81
19	7	NB	1	40.72
20	7	SVM	2	64.38
21	7	MaxEnt	3	32.69
22	8	NB	1	41.38
23	8	SVM	2	69.47
24	8	MaxEnt	3	33.25
25	9	NB	1	37.43
26	9	SVM	2	61.82
27	9	MaxEnt	3	32.31
28	10	NB	1	40.22
29	10	SVM	2	65.27
30	10	MaxEnt	3	33.04

## Appendix VIII: Comparison of means of the treatments

Comparison of mean			
F-score		Mean <sup>b</sup> (%)	N <sup>c</sup>
No. <sup>a</sup>	Treatments Name		
06	SS	33.55	2
18	SS + AFINN	38.87	2
12	SS + HL	42.20	2
16	SS + MPQA	44.66	2
03	MPQA	45.71	2
15	SS + HL + AFINN	45.78	2
02	HL	47.13	2
17	SS + MPQA + AFINN	47.78	2
29	SWN + SS + HL + AFINN	48.82	2
13	SS + HL + MPQA	48.86	2
09	HL + AFINN	49.63	2
14	SS + HL + MPQA + AFINN	50.05	2
04	AFINN	51.59	2
07	HL + MPQA	52.55	2
10	MPQA + AFINN	52.75	2
08	HL + MPQA + AFINN	55.43	2
20	SWN + HL + MPQA	60.82	2
19	SWN + HL	61.10	2
23	SWN + MPQA	61.13	2
05	SWN	61.25	2
25	SWN + AFINN	61.84	2
26	SWN + SS + HL	62.22	2
30	SWN + SS + MPQA	62.50	2
27	SWN + SS + HL + MPQA	62.63	2
31	SWN + SS + MPQA + AFINN	62.71	2
24	SWN + MPQA + AFINN	62.84	2
22	SWN + HL + AFINN +	62.91	2
32	SWN + SS + AFINN	62.93	2
11	SWN + SS	63.51	2
21	SWN + HL + MPQA + AFINN	63.80	2
28	SWN + SS + HL + MPQA + AFINN	64.27	2
50	TR + SWN + HL	77.67	2
40	TR + SS	80.23	2
41	TR + SWN	80.50	2
54	TR + SWN + MPQA	80.76	2
42	TR + SWN + SS	81.06	2
43	TR + SS + HL	81.25	2
61	TR + SWN + SS + MPQA	81.31	2
51	TR + SWN + HL + MPQA	81.48	2
49	TR + SS + AFINN	81.520	2
47	TR + SS + MPQA	81.59	2
58	TR + SWN + SS + HL + MPQA	81.65	2
33	TR + HL	81.66	2
57	TR + SWN + SS + HL	81.77	2
01	TR	81.84	2
55	TR + SWN + MPQA + AFINN	81.90	2
56	TR + SWN + AFINN	81.90	2
46	TR + SS + HL + AFINN	81.95	2
63	TR + SWN + SS + AFINN	81.96	2

62	TR + SWN + SS + MPQA + AFINN	81.99	2
59	TR + SWN + SS + HL + MPQA + AFINN	82.01	2
34	TR + HL + MPQA	82.04	2
44	TR + SS + HL + MPQA	82.15	2
52	TR + SWN + HL + MPQA + AFINN	82.19	2
60	TR + SWN + SS + HL + AFINN	82.19	2
53	TR + SWN + HL + AFINN	82.27	2
37	TR + MPQA	82.28	2
48	TR + SS + MPQA + AFINN	82.33	2
<b>38</b>	<b>TR + MPQA + AFINN</b>	<b>82.47</b>	<b>2</b>
<b>36</b>	<b>TR + HL + AFINN</b>	<b>82.53</b>	<b>2</b>
<b>45</b>	<b>TR + SS + HL + MPQA + AFINN</b>	<b>82.59</b>	<b>2</b>
<b>35</b>	<b>TR + HL + MPQA + AFINN</b>	<b>82.75</b>	<b>2</b>
<b>39</b>	<b>TR + AFINN</b>	<b>82.96</b>	<b>2</b>
Total		68.3554	126

a. No. refers to the number of the treatment which starts from 1 to 63.  
b. Mean refers to the means of variables. In this case, they are the results of F-score (%).  
c. N refers to the number of treatment used with machine learning algorithms. They showed 2 because the results of them from SVM (Kecman, 2005) and NB (Tan *et al.*, 2009).



## **Appendix IX: Publications**

### **List of Publications**

- Publication 1: The 6th Conference on Software, Knowledge, Information Management and Applications (SKIMA 2012)
- Publication 2: The 4th International Conference on Computer Technology and Development (ICCTD 2012)
- Publication 3: International Journal of Innovation, Management and Technology (IJIMT 2013)
- Publication 4: The International Workshop on Semantic Evaluation (SemEval 2013)
- Publication 5: Advances in Computer and Electronics Technology (ACET 2013) Publication 6: The International Workshop on Semantic Evaluation (SemEval 2014)
- Publication 7: International Journal of Advances in Engineering and Technology (IJAET 2015)
- Publication 8: International Conference on Computer and Information Technology (ICCIT 2015)
- Publication 9: The International Conference on Information Science & Applications (ICISA 2015)

Publication 1: The 6th Conference on Software, Knowledge, Information Management and  
Applications (SKIMA 2012)

## Using SentiWordNet and Sentiment Analysis for Detecting Radical Content on Web Forums

Tawunrat Chalothorn and Jeremy Ellman

Computing, Engineering & Information Sciences, University of Northumbria at Newcastle,  
Newcastle Upon Tyne, United Kingdom

**Abstract**—The internet has become a major tool for communication, training, fundraising, media operations, and recruitment, and these processes often use web forums. This paper presents a model that was built using SentiWordNet, WordNet and NLTK to analyze selected web forums that included radical content. The approaches of the model measure and identify sentiment polarity and affect the intensity of that which appears in the web forum.

**Index Terms**—SentiWordNet, sentiment, analysis, web forums, radical

### Introduction

Web forums have become important places for social communication and discussion on the internet. Some radical groups also use them for communication and disseminating their ideologies to the public [1]. The terrorists' main goals in using the internet are often research, communication, training, fundraising, media operations, radicalization and recruitment [2]. This research presents the system approach of two web forums in the area of sentiment and affects analysis.

Many people have questioned why this research was carried out. The reason is that the United Kingdom's parliament has enacted an anti-terrorism law, the Terrorism Act 2006 [3 and 4], which extends the government's ability to outlaw terrorist organizations that promote and encourage or may be thought to encourage terrorism [5]. In 2007 they launched the 'Prevent Strategy' to prevent the radicalization of youths in Great Britain and block networks that support terrorists [6]. The internet has become the main tool used by terrorists since it can be accessed anywhere and it gives access to a wide spectrum of

ideological material that may be translated into multiple languages [7].

This paper is structured as follows: Section II provides some discussion on work related to sentiment analysis and SentiWordNet. SentiWordNet is a lexical resource that supports opinion mining by assigning a positivity score and a negativity score to each WordNet. Section III discusses the research question. Data collection and the system technique were described in section IV. Finally, results analysis are presented in sections V.

### Related work

The term 'sentiment' was used by [8] and [9] in reference to the automatic analysis of evaluative text, and the tracking of predictive judgments and analysis of market sentiment in [10]. After that, the term 'opinion mining' was brought to the WWW conference by [11]. They mentioned that the ideal opinion-mining tools would press a set of search results for a given item, generating a list of product attributes and aggregating opinions about each of them [10]. Sentiment analysis has been considered in many research fields, such as [12] where sentiment analysis was used to analyze video comments and user profiles. In [13], the structure of lexical contextual sentences was used to classify sentiment classification from online customer reviews. In [14], SentiWordNet was used for classifying movie reviews in German. In addition, SentiWordNet was used in [15] for sentiment classification of reviews. As far as we are concerned, there are some papers that have used data from websites, blogs and forums but they have conducted testing

using Machine Learning and there are no existing papers that have used data from radical web forums for testing with SentiWordNet.

### Research Questions

Opinions and emotions are used on the internet for communication and can be related to and involve radical ideologies. This paper presents our research on sentiment analysis and the detection of radical content. In particular, this research analyzes an existing technique in an attempt to answer the research question ‘How effective is SentiWordNet for detecting opinions and emotions on the internet?’

### Methods

Two forums were selected for use in the research: Montada and Qawem. Both of them use the Arabic language. 500 sentences of each forum were translated manually for use in the experiment. Model building was written using Python programming language. The model building phase was started by splitting sentences into words and reducing the high-frequency text (stopwords) in the sentences. Words were stored in a bag of words (BOW) and part of speech (POS) was used for tagging words and knowing the position of each word in the sentence. Lexicon, WordNet and SentiWordNet were used for assigning positive and negative scores of each synset in each word [7].

The formulas for calculating positive and negative scores were taken from [16], as shown in (1) and (2). The final scores of sentences were calculated using a formula taken from [8], as shown in (3). The scores of sentences were applied using the rule that if the sentence had a positive score more than or equal to its negative score, then the sentence would be classified as positive. Otherwise it would be negative.

$$Pos\_weight = \left[ \frac{pos}{senses} \right]$$

$$Neg\_weight = \left[ \frac{neg}{senses} \right]$$

$pos$  is the number of lemma that have  $Pos(s)(i) \geq Neg(s)(i)$  and  $Pos(s)(i) \neq$

0;  $neg$  is the number of lemma that have  $Neg(s)(i) \geq Pos(s)(i)$  and  $Neg(s)(i) \neq 0$ ; and  $senses$  is the total number of lemma in synsets.

$$Sentence\_score = \left[ \frac{\sum_{i=1}^n Score(i)}{n} \right]$$

$Sentence\_score$  is the positive or negative scores of sentences;  $Score(i)$  is the positive or negative scores of the word in sentences; and  $n$  is the number of words in sentences.

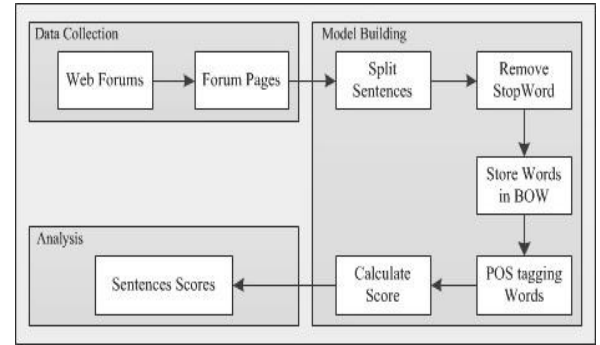


Fig. 1. Overall process of the system

### Results

The model building of sentiment was applied to the web forums Montada and Qawem for analysis of the results. After removing stopwords, the rest of the sentences were used for analysis. The search function in the system was used to extract statistics of corpus for getting information about the frequency of words that were used in the forums. The content in the forums was expected to be manipulated by religion and ideology. In the comparison between Qawem and Montada, it was found that Qawem contained more words related to radical ideology than Montana. In the results of the sentiment analysis of postings as percentages show that the Montada forum has less negative postings than the Qawem forum. In particular, the radical affect is quite strong in the communication found in the Qawem forum.

### Conclusion

In this research we have presented an analysis of two web forums, Montada and Qawem. The approach of model building and the results were explained. Overall, the

results show that Qawem has more radical content than Montada. For future work, a comparative human evaluation can take place. We will ask people to rate sentences and see how their opinions on a rating scale compare to those of the model.

## References

- [1] J. Glaser, J. Dixit, and D. P. Green, "Studying Hate Crime with the Internet: What Makes Racists Advocate Racial Violence?" *Journal of Social Issues*, vol. 58, pp. 177-193, 2002.
- [2] R. Tong, "An operational system for detecting and tracking opinions online," presented at the In Proc of SIGR Workshop on Operational Text Classification, New Orleans, Louisiana, 2001.
- [3] HM Government. (2006). Terrorism Act 2006. Available: [www.legislation.gov.uk/ukpga/2006/11/section/1](http://www.legislation.gov.uk/ukpga/2006/11/section/1)
- [4] J. C. Paye and J. H. Membrez, *Global war on liberty*: Telos Press Pub., 2007.
- [5] E. Parker, "Implementation of the UK Terrorism Act 2006 - The Relationship between Counterterrorism Law, Free Speech, and the Muslim Community in the United Kingdom versus the United States," *Emory International Law Review* 21 *Emory Int'l L. Rev.*, pp. 711-758, 2007.
- [6] G. Morgan, "Government to block terrorist web sites," in *computing.co.uk*, ed, 2011.
- [7] P. Nesserera, "Ideologies of Jihad in Europe," *Terrorism and Political Violence*, vol. 23, pp. 173-200, 2011.
- [8] S. Das and M. Chen, "Yahoo! for Amazon: Extracting market sentiment from stock message boards," presented at the Proceedings of the Asia Pacific Finance Association Annual Conference APFA, 2001.
- [9] R. Tong, "An operational system for detecting and tracking opinions online," presented at the In Proc of SIGR Workshop on Operational Text Classification, New Orleans, Louisiana, 2001.
- [10] B. Pang and L. Lee, "Opinion Mining and Sentiment Analysis," *Found. Trends Inf. Retr.*, vol. 2, pp. 1-135, 2008.
- [11] K. Dave, S. Lawrence, and D. M. Pennock, "Mining the peanut gallery: opinion extraction and semantic classification of product reviews," presented at the Proceedings of the 12th International Conference on the World Wide Web, Budapest, Hungary, 2003.
- [12] A. Bermingham, M. Conway, L. McInerney, N. O'Hare, and A. F. Smeaton, "Combining Social Network Analysis and Sentiment Analysis to Explore the Potential for Online Radicalisation," presented at the Proceedings of the 2009 International Conference on Advances in Social Network Analysis and Mining, 2009.
- [13] A. Khan and B. Baharudin, "Sentiment classification using sentence-level semantic orientation of opinion terms from blogs," presented at the National Postgraduate Conference (NPC), 2011.
- [14] K. Denecke, "Using SentiWordNet for multilingual sentiment analysis," presented at the Data Engineering Workshop, 2008. ICDEW 2008. IEEE 24th International Conference, 2008.
- [15] A. Hamouda and M. Rohaim, "Reviews Classification Using SentiWordNet Lexicon," *The Online Journal on Computer Science and Information Technology (OJCSIT)* vol. 2, pp. 120-123, 2011.

Publication 2: The 4th International Conference on Computer Technology and Development  
(ICCTD 2012)

**SENTIMENT ANALYSIS OF WEB FORUMS:  
COMPARISON BETWEEN SENTIWORDNET AND SENTISTRENGTH**

**TAWUNRAT CHALOTHORN**

Computing, Engineering & Information Sciences  
University of Northumbria,  
Newcastle Upon Tyne  
tawunrat.chalothorn@unn.ac.uk

**JEREMY ELLMAN**

Computing, Engineering & Information Sciences  
University of Northumbria,  
Newcastle Upon Tyne  
jeremy.ellman@unn.ac.uk

**ABSTRACT**

Internet has become a major tool for communication, training, fundraising, media operations, and recruitment, and these processes often use web forums. This paper intended to find suitable technique for analysing selected web forums that included radical content by presenting a comparison between SentiWordNet and SentiStrength. SentiWordNet is a lexical resource for supporting opinion mining by assigning a positivity score and a negativity score to each WordNet. SentiStrength is a technique that was developed from comments on MySpace. It uses human-designed lexical and emotional terms with a set of amplification, diminishing and negation rules. The results have been presented and discussed.

**KEY WORDS**

SentiWordNet, SentiStrength, sentiment, analysis, web forums, radical

## 1 Introduction

Web forums have become important places for social communication and discussion on the internet. Some radical groups also use them for communication and disseminating their ideologies to the public [1]. These kinds of forums can be referred to as part of the Dark Web. The Dark Web includes websites that are used by terrorists, radicals and extremist groups [2]. This paper presents the two system approach of two web forums in the area of sentiment and affects analysis. Their content is related to radicalization. The sections of this paper are structured as follows: section 2 provides some discussion on work related to sentiment analysis, SentiWordNet and SentiStrength. SentiWordNet is a lexical resource for supporting opinion mining by assigning a positivity score and a negativity score to each WordNet. SentiStrength is a sentiment analysis technique that was developed from comments on MySpace. It uses human-designed lexical and emotional terms [3]. Section 3 discusses the research question and this is followed by details of the data collection in section 4. System techniques were developed to assign and measure the effect of and sentiment found in the communication of web forums, as described in section 5. Finally, the analyses of the results are presented in section 6.

## 2 Related work

The term ‘sentiment’ was used by [4] and [5] in reference to the automatic analysis of evaluative text and tracking of predictive judgements, as well as analysing market sentiment [6]. Afterwards, the term ‘opinion mining’ was used at a WWW conference by [7]. They mentioned that the ideal opinion mining tools would press a set of search results for a given item, generating a list of product attributes and aggregation opinions about each of them [6]. Sentiment analysis has been used in many research fields, such as [8] who used sentiment analysis to analyse video comments and user profiles. [9] used the structure of lexical contextual sentences to classify sentiment from online customer reviews. Moreover, there are some researches that have used SentiWordNet and SentiStrength for classifying content, whether positive or negative. For instance, SentiWordNet was used for determining the polarity of reviews within the English and German languages by [10], and to classify movie reviews by [11]. [12] used SentiStrength to detect comments on MySpace. Also, SentiStrength was used for classifying emotions within reviews and analysing the content of Twitter by [3] and [13], respectively.

## 3 Research questions

Web forums have become the main tool for communicating with others as they can be accessed anywhere. Sometimes they are used by a group of people who have radical ideologies for research, communication, training, fundraising, media operations, radicalisation and recruitment [14]. This paper presents our research on sentiment analysis and detection of radical content. In particular, this research attempts to answer the research question ‘which technique of sentiment analysis can be used for classifying radical contents on web forums?’

#### 4 Data

Two forums have been selected for using in the research: Montada and Qawem. Both of them use the Arabic language. They have been selected by using research 21 people who are Arabic speaker by asking them that which websites they think that might have the contents related to radical Islamic ideologies. The results showed that Qawem and Montada are in the highest range.

#### 5 Methods

Data has been collected from the two web forums and classification of polarity has taken place using two techniques of sentiment analysis: SentiWordNet and SentiStrength. The model building phase when using SentiWordNet was started by splitting sentences into words and reducing the high-frequency text (stopwords) in sentences. Words were stored in a bag of words (BOW) and part of speech (POS) was used for tagging words and knowing the position of each word in the sentence. Lexicon, WordNet and SentiWordNet were used for assigning positive and negative scores for each synset in each word [8]. The formulas for calculating positive and negative scores were taken from [15]. The final scores of sentences were calculated using a formula taken from [9]. The scores of sentences were applied using the rule that if the sentence had a positive score more than or equal to its negative score, then the sentence would be classified as positive. Otherwise, it would be negative. An objective (neutral) score was not used. The sum of positive, negative and objective was equal to 1.0. After that, the technique SentiStrength was applied for classifying the data on a scale from 1 to 5: 1 meant that there was no sentiment and 5 meant that there was a very strong positive or negative sentiment [12]. The overall results from SentiStrength were based on the formula shown in (1).

$$\left. \begin{array}{ll} \text{if } \text{positive} > \text{negative}; & \text{Positive Sentiment} \\ \text{if } \text{positive} < \text{negative}; & \text{Negative Sentiment} \\ \text{if } \text{positive} = \text{negative}; & \text{Neutrality Sentiment} \end{array} \right\} (1)$$

#### 6 Results

Model building of sentiment was applied to the web forums Montada and Qawem so as to analyse the results. After removing stopwords, the rest of the sentences were used for analysis using the technique SentiWordNet, while the full sentences were used for analysis using SentiStrength without removing any words. The results show that the Montada forum has less negative postings than the Qawem forum. In particular, the radical effect is quite strong in the communication found in the Qawem forum. Nearly 35% of the postings in Qawem were found to have a negative score between 0.050 and 0.100, while Montada had less than 15% of postings in the same score range when using SentiWordNet. When using SentiStrength it was found that nearly 50% of the postings in Qawem had a negative score at 2, while only 30% of the postings in Montada had the same score. On the other hand, using SentiWordNet it was found that the positive scores of postings in the Montada forum were higher than those in the Qawem forum, except in the range from 0.100 to 0.150. Using SentiStrength it was found that the positive scores of postings in Montada were higher than in Qawem in every range from 1 to 5. From the overall results it can be seen that both techniques seem to work well for classifying the content of web forums. However, there are some problems if checking the score of sentences one by one.

#### 7 Conclusion

In this paper we have presented an analysis of two web forums, Montada and Qawem. They were chosen because their content relates to radicalization. The results of a comparison between SentiWordNet and SentiStrength were presented. The overall results of both techniques showed that Qawem had a higher percentage of postings with negative sentences than Montada. This said,

both techniques could be used for classifying the content of web forums. However, when checking scores of each sentence, there were incorrect scores in some sentences. The reason might be that, when using SentiWordNet, stopwords were removed from sentences and some words with negative meanings did not have a strong negative score, such as traitor and kill. This might have affected the meaning and the score of the sentence. For example, “My avenger” gives a different meaning to the sentence than “Avenger”, removing the stopword “My”. “My avenger” would have had a higher negative score than “Avenger”. Another sentence, “God cleans Syria from the traitors”, should have had a negative score instead of a positive score. The sentence aims to encourage people to fight in Syria, which is obviously radical in nature. Therefore, it would be better if SentiWordNet were to score stopwords and it should review the scores of some words that are negative. On the other hand, some incorrect scores occur when using SentiStrength, such as “Shiites”. “Shiites” should get a positive score instead of a negative score because the word refers to a group of people who believe in Islam and Ali [16,17 and 18]. This could be the reason why the sentence “God blesses Shiites everywhere” got a negative score instead of a positive score. Also, in the second sentence “armed” and “liberate” are negative in sentiment but SentiStrength showed that they were neutral. The reason for them getting a neutrality score might be that the words are not in their database. The methodology of SentiStrength was developed from comments on MySpace and such words may not have appeared in the comments. Therefore, there is a possibility of using SentiStrength as a model for developing another methodology for classifying and detecting the content of web forums, which will be part of our future work.

## References

- [1] Jack Glaser; Jay Dixit; Donald P .Green. Studying Hate Crime with the Internet: What Makes Racists Advocate Racial Violence? *Journal of Social Issues* 58 (1), 2002, PP: 177-193.
- [2] Hsinchun Chen. *Intelligence and Security Informatics For International Security: Information Sharing and Data Mining*. Springer, 2006.
- [3] David Garcia; Frank Schweitzer. Emotions in Product Reviews--Empirics and Models. Paper presented at the Privacy, security, risk and trust (passat), IEEE 3rd international conference on social computing (socialcom), 2011, PP: 483-488.
- [4] Sanjiv Das; Mike Chen. Yahoo! for Amazon: Extracting market sentiment from stock message boards. Paper presented at the Proceedings of the Asia Pacific Finance Association Annual Conference APFA, 2001, PP: 37-56.
- [5] Richard Tong. An operational system for detecting and tracking opinions in on-line. Paper presented at the In Proc of SIGR Workshop on Operational Text Classification, New Orleans, Louisiana, 2001, PP: 1-6.
- [6] Bo Pang; Lillian Lee. Opinion Mining and Sentiment Analysis. *Found Trends Inf Retr* 2 (1-2), 2008, PP: 1-135.
- [7] Kushal Dave; Steve Lawrence; David M. Pennock. Mining the peanut gallery: opinion extraction and semantic classification of product reviews. Paper presented at the Proceedings of the 12th international conference on World Wide Web, Budapest, Hungary, 2003, PP: 519-528.
- [8] Adam Bermingham; Maura Conway; Lisa McInerney; Neil O'Hare; Alan F. Smeaton. Combining Social Network Analysis and Sentiment Analysis to Explore the Potential for Online Radicalisation. Paper presented at the Proceedings of the 2009 International Conference on Advances in Social Network Analysis and Mining, 2009.
- [9] Aurangzeb Khan; Baharum Baharudin. Sentence Level Semantic Orientation of Online Reviews and Blogs using SentiWordNet for Effective Sentiment Classification. *International Journal of New Computer Architectures and their Applications (IJNCAA)* 1 (2), 2011, PP: 627-643.
- [10] Kerstin Denecke. Using SentiWordNet for multilingual sentiment analysis. Paper presented at the Data Engineering Workshop, 2008. ICDEW 2008. IEEE 24th International Conference, 2008, PP: 507-512.
- [11] Bruno Ohana; Brendan Tierney. Sentiment classification of reviews using SentiWordNet. Paper presented at the 9th. IT&T Conference, Dublin Institute of Technology, Dublin,

Ireland, 2009.

- [12] Mike Thelwall; Kevan Buckley; Georgios Paltoglou. Sentiment in Twitter events. *Journal of the American Society for Information Science and Technology* 62 (2), 2011, PP: 406-418.
- [13] Rene Pfitzner; Antonios Garas; Frank Schweitzer. Emotional Divergence Influences Information Spreading in Twitter. Paper presented at the The 6th International AAAI Conference on Weblogs and Social Media, Dublin, Ireland, 2012.
- [14] Barbara Mantel. Terrorism and the Internet : should Web sites that promote terrorism be shut down? In. Washington, D.C. : CQ Press, 2009, PP: 129-155.
- [15] Alena Neviarouskaya; Helmut Prendinger; Mitsuru Ishizuka. Textual Affect Sensing for Sociable and Expressive Online Communication. Paper presented at the Proceedings of the 2nd international conference on Affective Computing and Intelligent Interaction, Lisbon, Portugal, 2007, PP: 218-229.
- [16] George W. Braswell. What You Need to Know About Islam and Muslims. B&H Publishing Group, 2000.
- [17] L.R. Reddy. Inside Afghanistan: End of the Taliban Era? APH Publishing, 2002.
- [18] Heinz Halm. The Shiites: A Short History. Markus Wiener Publishers, 2007.



## **Affect Analysis of Radical Contents on Web Forums Using SentiWordNet**

Tawunrat Chalothorn and Jeremy Ellman

Computing, Engineering & Information Sciences, University of Northumbria at  
Newcastle Newcastle Upon Tyne, United Kingdom

**Abstract.** The internet has become a major tool for communication, training, fundraising, media operations, and recruitment, and these processes often use web forums. This paper presents a model that was built using SentiWordNet, WordNet and NLTK to analyze selected web forums that included radical content. SentiWordNet is a lexical resource for supporting opinion mining by assigning a positivity score and a negativity score to each WordNet. The approaches of the model measure and identify sentiment polarity and affect the intensity of that which appears in the web forum. The results show that SentiWordNet can be used for analyzing sentences that appear in web forums.

**Keywords:** SentiWordNet, sentiment, analysis, web forums, radical

### **1. Introduction**

Web forums have become important places for social communication and discussion on the internet. Some radical groups also use them for communication and disseminating their ideologies to the public [1]. These kinds of forums can be referred to as part of the Dark Web. The Dark Web includes websites that are used by terrorists, radicals and extremist groups [2]. This paper presents the system approach of two web forums in the area of sentiment and affects analysis. Their content is related to radicalization. Many people have questioned why this research was carried out. The reason is that the United Kingdom's parliament has enacted an anti-terrorism law, the Terrorism Act 2006 [3 and 4], which extends the government's ability to outlaw terrorist organizations that promote and encourage or may be thought to encourage terrorism [5]. In 2007 they launched the 'Prevent Strategy' to prevent the radicalization of youths in Great Britain and block networks that support terrorists [6]. The internet has become the main tool used by terrorists since it can be accessed anywhere and it gives access to a wide spectrum of ideological material that may be translated into multiple languages [7]. Their main goals in using the internet are often research, communication, training, fundraising, media operations, radicalization and recruitment [8].

This paper is structured as follows: Section 2 provides some discussion on work related to sentiment analysis and SentiWordNet. SentiWordNet is a lexical resource that supports opinion mining by assigning a positivity score and a negativity score to each WordNet. Section 3 discusses the research question and this is followed by details of the data collection in section 4. The system technique was developed to assign and measure the affect and sentiment found in the communication of web forums, as described in section 5. Finally, methods of model building and results analyses are presented in sections 6.

## 2. Related Work

The term ‘sentiment’ was used by [9] and [10] in reference to the automatic analysis of evaluative text, and the tracking of predictive judgments and analysis of market sentiment in [11]. After that, the term ‘opinion mining’ was brought to the WWW conference by [12]. They mentioned that the ideal opinion-mining tools would press a set of search results for a given item, generating a list of product attributes and aggregating opinions about each of them [11]. Sentiment analysis has been considered in many research fields, such as [13] where sentiment analysis was used to analyze video comments and user profiles. In [14], the structure of lexical contextual sentences was used to classify sentiment classification from online customer reviews. In [15], SentiWordNet was used for classifying movie reviews in German. In addition, SentiWordNet was used in [16] for sentiment classification of reviews. As far as we are concerned, there are some papers that have used data from websites, blogs and forums but they have conducted testing using Machine Learning and there are no existing papers that have used data from radical web forums for testing with SentiWordNet.

## 3. Research Question

The internet has become the main tool of radicals, extremists and terrorists since it can be accessed anywhere and allows access to a wide spectrum of ideological material that can be translated into multiple languages [7]. Opinions and emotions are used on the internet for communication and can be related to and involve radical ideologies. The terrorists' main goals in using the internet are often research, communication, training, fundraising, media operations, radicalization and recruitment [8]. This paper presents our research on sentiment analysis and the detection of radical content. In particular, this research analyzes an existing technique in an attempt to answer the research question ‘How effective is SentiWordNet for detecting opinions and emotions on the internet?’

## 4. Data

Two forums were selected for use in the research: Montada and Qawem. Both of them use the Arabic language. They were selected by asking 21 people who are Arabic speakers which websites they think might have content related to radical Islamic ideologies. The results showed that Qawem and Montada are in the highest range.

## 5. Methods

The overall process consisted of data collection, model building and result analysis, as shown in Fig. 1. The data collection phase has been described in the previous section. After that, 500 sentences of each forum were translated manually for use in the experiment. Model building was written using Python programming language. The model building phase was started by splitting sentences into words and reducing the high-frequency text (stopwords) in the sentences. Samples of stopwords can be found in Table 1. Words were stored in a bag of words (BOW) and part of speech (POS) was used, as shown in Table 2, for tagging words and knowing the position of each word in the sentence. Lexicon, WordNet and SentiWordNet were used for assigning positive and negative scores of each synset in each word [13].

The formulas for calculating positive and negative scores were taken from [17], as shown in (1) and (2). The final scores of sentences were calculated using a formula taken from [14], as shown in (3). The scores of sentences were applied using the rule that if the

sentence had a positive score more than or equal to its negative score, then the sentence would be classified as positive. Otherwise it would be negative. Example of sentences can be found in Table 3.

$$Pos\_weight = \left[ \frac{pos}{senses} \right] \quad (1)$$

$$Neg\_weight = \left[ \frac{neg}{senses} \right] \quad (2)$$

$pos$  is the number of lemma that have  $Pos(s)(i) \geq Neg(s)(i)$  and  $Pos(s)(i) \neq 0$ ;  $neg$  is the number of lemma that have  $Neg(s)(i) \geq Pos(s)(i)$  and  $Neg(s)(i) \neq 0$ ; and  $senses$  is the total number of lemma in synsets.

$$Sentence\_score = \left[ \frac{\sum_{i=1}^n Score(i)}{n} \right] \quad (3)$$

$Sentence\_score$  is positive or negative or negative scores of sentences;  $Score(i)$  is the positive or negative scores of the word in sentences; and  $n$  is the number of words in sentences.

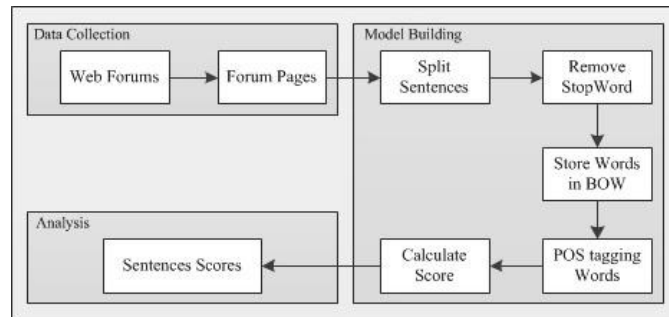


Fig. 1. Overall process of the system

Table 1. Samples of Stopwords

Stopwords
['i', 'me', 'my', 'myself', 'we', 'our', 'ours', 'ourselves', 'you', 'your', 'yours', 'yourself', 'yourselves', 'he', 'him', 'his', 'himself', 'she', 'her', 'hers', 'herself', 'it', 'its', 'itself', 'they', 'them', 'their', 'theirs', 'themselves', 'what', 'which', 'who', 'whom', 'this', 'that', 'these', 'those', 'am', 'is', 'are', 'was', 'were', 'be', 'been', ...]

Table 2. Parts of Speech Labels

POS Meaning	POS Tag	SentiWordNet Tag
Verb	VB, VBD, VBG, VBN, VBP, VBZ	V

POS Meaning	POS Tag	SentiWordNet Tag
Noun(s)	NN, NNS, NNP, NNPS	N
Adverb(s)	RB, RBR, RBS	R
Adjective(s)	JJ, JJR, JJS	A

Table 3. Example of Sentences with Sentiment Polarity <sup>68</sup>

Arabic and English Translation	Sentiment Polarity	
	Positive	Negative
الله يلعن الوهابية والسلفية اعداء الدين Allah curse the Salafi and Wahhabi enemies of religion.	0.000	0.033
اللهم انزل سخطك على يهود آل خليفة Allah send down your wrath on the Jews of Al-Khalifa.	0.019	0.100

## 6. Result

The model building of sentiment was applied to the web forums Montada and Qawem for analysis of the results. After removing stopwords, the rest of the sentences were used for analysis. The search function in the system was used to extract statistics of corpus for getting information about the frequency of words that were used in the forums, as shown in Fig. 2 and Fig. 3. The content in the forums was expected to be manipulated by religion and ideology. Both results showed that the top 10 most frequently used words were words related to religion, such as ‘God’ and ‘Allah’. ‘God’ was found to be the most frequently used word in both forums. In the comparison between Qawem and Montada, it was found that Qawem contained more words related to radical ideology than Montana, such as ‘curse’ and ‘enemies’. At the below, Fig. 4 and 5 show the results of the sentiment analysis of postings as percentages. The results show that the Montada forum has less negative postings than the Qawem forum. In particular, the radical affect is quite strong in the communication found in the Qawem forum. Nearly 35% of the postings in Qawem have a negative score between 0.050 and 0.100, while Montada has less than 15% of postings in the same score range. On the other hand, the positive scores of postings in the Montada forum were higher than those in the Qawem forum, except in the range from 0.100 to 0.150.

<sup>68</sup> These are not views expressed or implied by the author or the University of Northumbria at Newcastle.

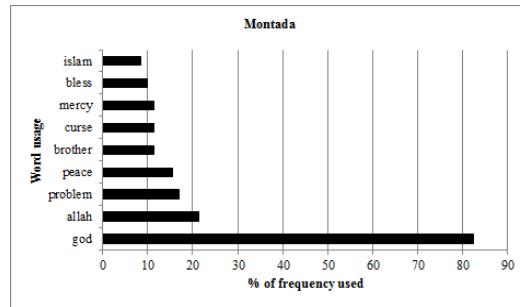


Fig. 2. Top high frequency words in Montada

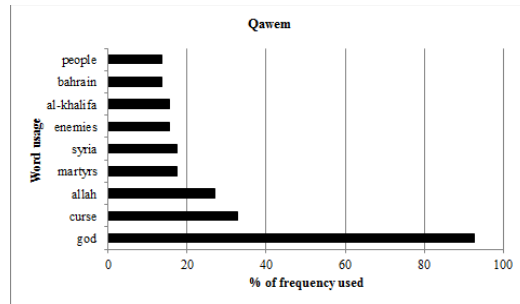


Fig. 3. Top high frequency words in Qawem

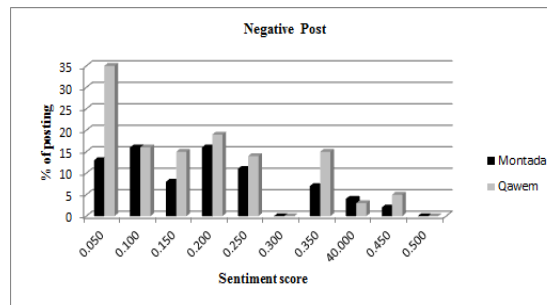


Fig. 4. Negative scores of sentiment analysis

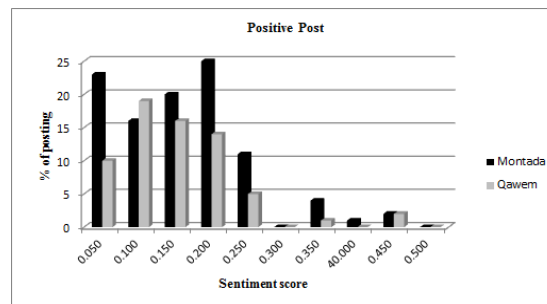


Fig. 5. Positive scores of sentiment analysis

## 7. Conclusion

In this paper we have presented an analysis of two web forums, Montada and Qawem. They were chosen because their content relates to radicalization. The approach of model building and the results were explained. The system was developed using SentiWordNet, WordNet and NLTK for analysis of data. Overall, the results show that Qawem has more radical content than Montada. For future work, a comparative human

evaluation can take place. We will ask people to rate sentences and see how their opinions on a rating scale compare to those of the model. Moreover, other techniques of sentiment analysis, such as SentiFul and SentiStrength, will be used for analyzing radical content. The aim will be to find suitable techniques for use in a model to be developed in the future.

## 8. References

- [1] J. Glaser, J. Dixit, and D. P. Green, "Studying Hate Crime with the Internet: What Makes Racists Advocate Racial Violence?" *Journal of Social Issues*, vol. 58, pp. 177-193, 2002.
- [2] H. Chen, *Intelligence and Security Informatics For International Security: Information Sharing and Data Mining*: Springer, 2006.
- [3] HM Government. (2006). *Terrorism Act 2006*. Available: <http://www.legislation.gov.uk/ukpga/2006/11/section/1>
- [4] J. C. Paye and J. H. Membrez, *Global war on liberty*: Telos Press Pub., 2007.
- [5] E. Parker, "Implementation of the UK Terrorism Act 2006 - The Relationship between Counterterrorism Law, Free Speech, and the Muslim Community in the United Kingdom versus the United States," *Emory International Law Review* 21 *Emory Int'l L. Rev.*, pp. 711-758, 2007.
- [6] G. Morgan, "Government to block terrorist web sites," in *computing.co.uk*, ed, 2011.
- [7] P. Nessler, "Ideologies of Jihad in Europe," *Terrorism and Political Violence*, vol. 23, pp. 173-200, 2011.
- [8] B. Mantel, "Terrorism and the Internet: should web sites that promote terrorism be shut down?" ed: Washington, D.C. : CQ Press, 2009, pp. 129-155.
- [9] S. Das and M. Chen, "Yahoo! for Amazon: Extracting market sentiment from stock message boards," presented at the Proceedings of the Asia Pacific Finance Association Annual Conference APFA, 2001.
- [10] R. Tong, "An operational system for detecting and tracking opinions on-line," presented at the In Proc of SIGR Workshop on Operational Text Classification, New Orleans, Louisiana, 2001.
- [11] B. Pang and L. Lee, "Opinion Mining and Sentiment Analysis," *Found. Trends Inf. Retr.*, vol. 2, pp. 1-135, 2008.
- [12] K. Dave, S. Lawrence, and D. M. Pennock, "Mining the peanut gallery: opinion extraction and semantic classification of product reviews," presented at the Proceedings of the 12th International Conference on the World Wide Web, Budapest, Hungary, 2003.
- [13] A. Bermingham, M. Conway, L. McInerney, N. O'Hare, and A. F. Smeaton, "Combining Social Network Analysis and Sentiment Analysis to Explore the Potential for Online Radicalisation," presented at the Proceedings of the 2009 International Conference on Advances in Social Network Analysis and Mining, 2009.
- [14] A. Khan and B. Baharudin, "Sentiment classification using sentence-level semantic orientation of opinion terms from blogs," presented at the National Postgraduate Conference (NPC), 2011.
- [15] K. Denecke, "Using SentiWordNet for multilingual sentiment analysis," presented at the Data Engineering Workshop, 2008. ICDEW 2008. IEEE 24th International Conference, 2008.
- [16] A. Hamouda and M. Rohaim, "Reviews Classification Using SentiWordNet Lexicon," *The Online Journal on Computer Science and Information Technology (OJCSIT)* vol. 2, pp. 120-123, 2011.
- [17] A. Neviarouskaya, H. Prendinger, and M. Ishizuka, "Textual Affect Sensing for Sociable and Expressive Online Communication," presented at the Proceedings of the 2nd International Conference on Affective Computing and Intelligent Interaction, Lisbon, Portugal, 2007.

Publication 4: The International Workshop on Semantic Evaluation (SemEval 2013)

## **TJP: Using Twitter to Analyze the Polarity of Contexts**

**Tawunrat Chalothorn**

University of Northumbria at Newcastle  
Pandon Building, Camden Street  
Newcastle Upon Tyne, NE2 1XE, UK  
Tawunrat.chalothorn@unn.ac.uk

**Jeremy Ellman**

University of Northumbria at Newcastle  
Pandon Building, Camden Street  
Newcastle Upon Tyne, NE2 1XE, UK  
Jeremy.ellman@unn.ac.uk

### **Abstract**

This paper presents our system, TJP, which participated in SemEval 2013 Task 2 part A: Contextual Polarity Disambiguation. The goal of this task is to predict whether marked contexts are positive, neutral or negative. However, only the scores of positive and negative class will be used to calculate the evaluation result using F-score. We chose to work as ‘constrained’, which used only the provided training and development data without additional sentiment annotated resources. Our approach considered unigram, bigram and trigram using Naïve Bayes training model with the objective of establishing a simple-approach baseline. Our system achieved F-score 81.23% and F-score 78.16% in the results for SMS messages and Tweets respectively.

### **1 Introduction**

Natural language processing (NLP) is a research area comprising various tasks; one of which is sentiment analysis. The main goal of sentiment analysis is to identify the polarity of natural language text (Shaikh et al., 2007). Sentiment analysis can be referred to as opinion mining, as study peoples’ opinions, appraisals and emotions towards entities and events and their attributes (Pang and Lee, 2008). Sentiment analysis has become a popular research area in NLP with the purpose of identifying opinions or attitudes in terms of polarity.

This paper presents TJP, a system submitted to SemEval 2013 for Task 2 part A: Contextual Polarity Disambiguation (Wilson et al., 2013). TJP was focused on the ‘constrained’

task, which used only training and development data provided. This avoided both resource implications and potential advantages implied by the use of additional data containing sentiment annotations. The objective was to explore the relative success of a simple approach that could be implemented easily with open-source software.

The TJP system was implemented using the Python Natural Language Toolkit (NLTK, Bird et al., 2009). We considered several basic approaches. These used a preprocessing phase to expand contractions, eliminate stopwords, and identify emoticons. The next phase used supervised machine learning and n-gram features. Although we had two approaches that both used n-gram features, we were limited to submitting just one result. Consequently, we chose to submit a unigram based approach followed by naive Bayes since this performed better on the data.

The remainder of this paper is structured as follows: section 2 provides some discussion on the related work. The methodology of corpus collection and data classification are provided in section 3. Section 4 outlines details of the experiment and results, followed by the conclusion and ideas for future work in section 5.

### **2 Related Work**

The micro-blogging tool Twitter is well-known and increasingly popular. Twitter allows its users to post messages, or ‘Tweets’ of up to 140 characters each time, which are available for immediate download over

the Internet. Tweets are extremely interesting to marketing since their rapid public interaction can either indicate customer success or presage public relations disasters far more quickly than web pages or traditional media. Consequently, the content of tweets and identifying their sentiment polarity as positive or negative is a current active research topic. Emoticons are features of both SMS texts, and tweets. Emoticons such as :) to represent a smile, allow emotions to augment the limited text in SMS messages using few characters. Read (2005) used emoticons from a training set that was downloaded from Usenet newsgroups as annotations (positive and negative). Using the machine learning techniques of Naïve Bayes and Support Vector Machines Read (2005) achieved up to 70 % accuracy in determining text polarity from the emoticons used.

Go et al. (2009) used distant supervision to classify sentiment of Twitter, as similar as in (Read, 2005). Emoticons have been used as noisy labels in training data to perform distant supervised learning (positive and negative). Three classifiers were used: Naïve Bayes, Maximum Entropy and Support Vector Machine, and they were able to obtain more than 80% accuracy on their testing data.

Aisopos et al. (2011) divided tweets in to three groups using emoticons for classification. If tweets contain positive emoticons, they will be classified as positive and vice versa. Tweets without positive/negative emoticons will be classified as neutral. However, tweets that contain both positive and negative emoticons are ignored in their study. Their task focused on analyzing the contents of social media by using n-gram graphs, and the results showed that n-gram yielded high accuracy when tested with C4.5, but low accuracy with Naïve Bayes Multinomial (NBM).

### 3 Methodology

#### 3.1 Corpus

The training data set for SemEval was built using Twitter messages training and development data. There are more than 7000 pieces of context. Users usually use emoticons in their tweets; therefore, emoticons have been manually collected and labeled as positive and negative to provide some context (Table 1), which is the same idea as in Aisopos et al. (2011).

Negative emoticons	:( :-(: :d :< D: :\/: etc.
Positive emoticons	:) ;) :-) ;-) :P ;P (: (; :D ;D etc.

Table 1: Emoticon labels as negative and positive

Furthermore, there are often features that have been used in tweets, such as hashtags, URL links, etc. To extract those features, the following processes have been applied to the data.

1. Retweet (RT), twitter username (@panda), URL links (e.g. y2u.be/fiKKzdLQvFo), and special punctuation were removed.
2. Hashtags have been replaced by the following word (e.g. # love was replaced by love, # exciting was replaced by exciting).
3. English contraction of 'not' was converted to full form (e.g. don't -> do not).
4. Repeated letters have been reduced and replaced by 2 of the same character (e.g. happpppppy will be replaced by happy, coollllll will be replaced by cool).

#### 3.2 Classifier

Our system used the NLTK Naïve Bayes classifier module. This is a classification based on Bayes's rule and also known as the state-of-art of the Bayes rules (Cufoglu et al., 2008). The Naïve Bayes model follows the assumption that



attributes within the same case are independent given the class label (Hope and Korb, 2004).

Tang et al. (2009) considered that Naïve Bayes assigns a context  $X_i$  (represented by a vector  $X_i^*$ ) to the class  $C_j$  that maximizes  $P(C_j|X_i^*)$  by applying Bayes's rule, as in (1).

$$P(C_j|X_i^*) = \frac{P(C_j)P(X_i^*|C_j)}{P(X_i^*)} \quad (1)$$

where  $P(X_i^*)$  is a randomly selected context  $X$ . The representation of vector is  $X_j^*$ .  $P(C)$  is the random select context that is assigned to class  $C$ .

To classify the term  $P(X_i^*|C_j)$ , features in  $X_i^*$  were assumed as  $f_j$  from  $j = 1$  to  $m$  as in (2).

$$P(C_j|X_i^*) = \frac{P(C_j) \prod_{j=1}^m P(f_j|C_j)}{P(X_i^*)} \quad (2)$$

There are many different approaches to language analysis using syntax, semantics, and se-mantic resources such as WordNet. That may be exploited using the NLTK (Bird et al. 2009). However, for simplicity we opted here for the n-gram approach where texts are decomposed into term sequences. A set of single sequences is a unigram. The set of two word sequences (with overlapping) are bigrams, whilst the set of overlapping three term sequences are trigrams. The relative advantage of the bi-and trigram approaches are that coordinates terms effectively disambiguate senses and focus content retrieval and recognition.

N-grams have been used many times in contents classification. For example, Pang et al. (2002) used unigram and bigram to classify movie reviews. The results showed that unigram gave better results than bigram. Conversely, Dave et al. (2003) reported gaining better results from trigrams rather than bigram in classifying product reviews. Consequently, we chose to evaluate

unigrams, bigrams and trigrams to see which will give the best results in the polarity classification. Our results are described in the next section.

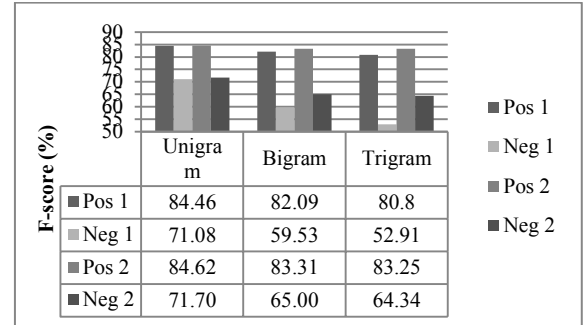


Figure 1: Comparison of Twitter messages from two approaches

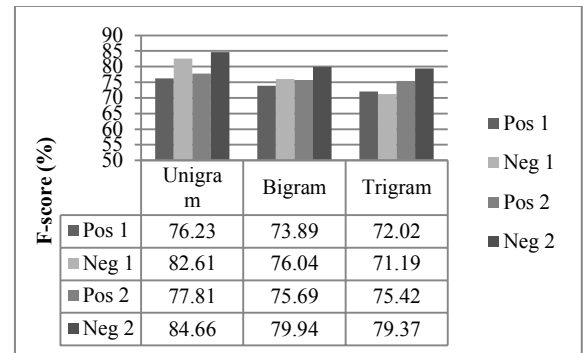


Figure 2: Comparison of SMS messages from two approaches

## 4 Experiment and Results

In this experiment, we used the distributed data from Twitter messages and the F-measure for system evaluation. As at first approach, the corpora were trained directly in the system, while stopwords (e.g. a, an, the) were removed before training using the python NLTK for the second approach. The approaches are demonstrated on a sample context in Table 2 and 3.

After comparing both approaches (Figure 1), we were able to obtain an F-score 84.62% of positive and 71.70% of negative after removing stopwords. Then, the average F-score is 78.16%, which was increased from the first approach by 0.50%. The results from both approaches showed that, unigram

achieved higher scores than either bigrams or trigrams.

Moreover, these experiments have been tested with a set of SMS messages to assess how well our system trained on Twitter data can be generalized to other types of message data. The second approach still achieved the better scores (Figure 2), where we were able to obtain an F-score of 77.81% of positive and 84.66% of negative; thus, the average F-score is 81.23%.

The results of unigram from the second approach submitted to SemEval 2013 can be found in Figure 3. After comparing them using the average F-score from positive and negative class, the results showed that our system works better for SMS messaging than for Twitter.

gonna miss some of my classes.		
Unigram	Bigram	Trigram
gonna miss some of my classes	gonna miss miss some some of of my my classes	gonna miss some miss some of some of my of my classes

Table 2: Example of context from first approach

gonna miss ( <i>some of</i> ) my classes.		
Unigram	Bigram	Trigram
gonna miss my classes	gonna miss miss my my classes	gonna miss my miss my classes

Table 3: Example of context from second approach. Note ‘some’ and ‘of’ are listed in NLTK stopwords.

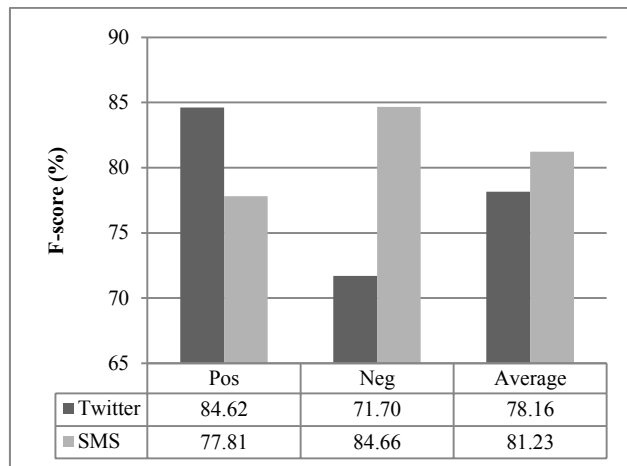


Figure 3: Results of unigram of Twitter and SMS in the second approach

## 5 Conclusion and Future Work

A system, TJP, has been described that participated in SemEval 2013 Task 2 part A: Contextual Polarity Disambiguation (Wilson et al., 2013). The system used the Python NLTK (Bird et al 2009) Naive Bayes classifier trained on Twitter data. Furthermore, emoticons were collected and labeled as positive and negative in order to classify contexts with emoticons. After analyzing the Twitter message and SMS messages, we were able to obtain an average F-score of 78.16% and 81.23% respectively during the SemEval 2013 task. The reason that, our system achieved better scores with SMS message than Twitter message might be due to our use of Twitter messages as training data. However this is still to be verified experimentally.

The experimental performance on the tasks demonstrates the advantages of simple approaches. This provides a baseline performance set to which more sophisticated or resource intensive techniques may be compared.

For future work, we intend to trace back to the root words and work with the suffix and prefix that imply negative semantics, such as ‘dis-’, ‘un-’, ‘-ness’

and ‘-less’. Moreover, we would like to collect more shorthand texts than that used commonly in microblogs, such as gr8 (great), btw (by the way), pov (point of view), gd (good) and nel (anyone). We believe these could help to improve our system and achieve better accuracy when classifying the sentiment of context from microblogs.

## References

- Alec Go, Richa Bhayani and Lei Huang. 2009. Twitter sentiment classification using distant supervision. *CS224N Project Report*, Stanford, 1-12.
- Ayşe Cufoglu, Mahi Lohi and Kambiz Madani. 2008. *Classification accuracy performance of Naive Bayesian (NB), Bayesian Networks (BN), Lazy Learning of Bayesian Rules (LBR) and Instance-Based Learner (IB1) - comparative study*. Paper presented at the Computer Engineering & Systems, 2008. ICCES 2008. International Conference on.
- Bo Pang, Lillian Lee and Shivakumar Vaithyanathan. 2002. *Thumbs up?: sentiment classification using machine learning techniques*. Paper presented at the Proceedings of the ACL-02 conference on Empirical methods in natural language processing - Volume 10.
- Fotis Aisopos, George Papadakis and Theodora Varvarigou. 2011. *Sentiment analysis of social media content using N-Gram graphs*. Paper presented at the Proceedings of the 3rd ACM SIGMM international workshop on Social media, Scottsdale, Arizona, USA.
- Huifeng Tang, Songbo Tan and Xueqi Cheng. 2009. A survey on sentiment detection of reviews. *Expert Systems with Applications*, 36(7), 10760-10773.
- Jonathon. Read. 2005. *Using emoticons to reduce dependency in machine learning techniques for sentiment classification*. Paper presented at the Proceedings of the ACL Student Research Workshop, Ann Arbor, Michigan.
- Kushal Dave, Steve Lawrence and David M. Pennock. 2003. *Mining the peanut gallery: opinion extraction and semantic classification of product reviews*. Paper presented at the Proceedings of the 12th international conference on World Wide Web, Budapest, Hungary.
- Lucas R. Hope and Kevin B. Korb. 2004. *A bayesian metric for evaluating machine learning algorithms*. Paper presented at the Proceedings of the 17th Australian joint conference on Advances in Artificial Intelligence, Cairns, Australia.
- Mostafa Al Shaikh, Helmut Prendinger and Ishizuka Mitsuru. 2007. *Assessing Sentiment of Text by Semantic Dependency and Contextual Valence Analysis*. Paper presented at the Proceedings of the 2nd international conference on Affective Computing and Intelligent Interaction, Lisbon, Portugal.
- Pang Bo and Lillian Lee. 2008. Opinion Mining and Sentiment Analysis. *Found. Trends Inf. Retr.*, 2(1-2), 1-135.
- Steven Bird, Ewan Klein and Edward Loper. 2009. *Natural language processing with Python*: O'Reilly.
- Theresa Wilson, Zornitsa Kozareva, Preslav Nakov, Sara Rosenthal, Veselin Stoyanov and Alan Ritter. 2013. SemEval-2013 Task 2: Sentiment Analysis in Twitter *Proceedings of the 7th International Workshop on Semantic Evaluation: Association for Computational Linguistics*.

Publication 5: Advances in Computer and Electronics Technology (ACET 2013)

# Sentiment Analysis: State of the Art

[Tawunrat Chalothorn and Jeremy Ellman]

**Abstract—** We presented the state of art of sentiment analysis which contained about the purpose of sentiment analysis, levels of sentiment analysis and processes that could be used to measure polarity and classify labels. Moreover, brief details about some resources of sentiment analysis are included.

**Keywords—** sentiment, analysis, natural language processing, nlp, sentiwordnet,

## Introduction

Sentiment can be defined as a tendency to experience certain emotions in relation to a particular object or person (Leuba, 1961; Richmond, 1965). Sentiment is expressed usually in writing, such as products review, websites, blogs, forums, etc. Sometimes, opinions are hidden within long sentences, making them difficult to reads and extract.

There is a technique called, ‘sentiment analysis’ that relates to natural language processing, text mining and linguistics (Hogenboom *et al.*, 2012). The main goal of sentiment analysis is to identify the polarity of natural language text (Shaikh *et al.*, 2007), which is not limited to positive and negative (Karlgrén *et al.*, 2012). Sentiment analysis can be referred to as opinion mining as both study people’s opinions, appraisals and emotions towards entities, events and their attributes (Pang and Lee, 2008).

The following section contains examples of some languages to which sentiment analysis has been applied.

(Waltinger, 2010) developed a lexicon resource in German called, GermanPolarityClues. It was created using a combination of a semi-automatic translation method and a manual assessment and extension of individual polarity-based term features. The results demonstrated that GermanPolarityClues attained performance of 87.6% F1-measure. F1-measure is an average precision and recall used frequently to measure the overall performance of the method. More details of this can be found in (Rijsbergen, 1979).

(Haruechaiyasak *et al.*, 2010) have used sentiment analysis to develop Thai resources for classifying hotel reviews by creating their own domain-independent lexicons. (Kongthon *et al.*, 2011) extended the previous work of (Haruechaiyasak *et al.*, 2010) by using features and polar words based on syntactic pattern analysis. From this, they constructed a Thai lexicon by increasing the data to approximately 12,000 reviews, covering 620 hotels. Their tasks achieved between 85% and 87% F1-measure.

(Wan, 2008) used both Chinese and English lexicons to improve sentiment analysis in Chinese. (Ku *et al.*, 2009) analyzed Chinese opinion by using the structure of Chinese words. They tested 4 tasks: word extraction; word polarity detection; sentence extraction; and sentence polarity detection. The results showed that they obtained the highest score for sentence extraction at 80% F1-measure and the lowest score at 54% F1-measure for sentence polarity detection.

(Ghorbel and Jacot, 2011) analyzed French movie reviews using the lexicon-based method, SentiWordNet, part-of-speech and stopwords. They translate from French to English using SentiWordNet for polarity extraction. From their experiments, they achieved over 85% accuracy.

(Kieu and Pham, 2010) developed a system to analyze product reviews for Vietnamese at sentence level. There is no public corpus available for Vietnamese sentiment analysis; therefore, they have to use GATE to create their own rule-based system. GATE is an open source software for use in text processing, more details of which can be found in (Cunningham *et al.*, 2011). Data was collected from an online product-advertising page featuring two categories - laptops and desktops – and 3,971 sentences. An annotation tool called Callisto (Day *et al.*, 2004) has been used to amend their corpus. They used GATE JAPE Grammar (Thakker *et al.*, 2009) to specify their rules, which can be divided into four

types: dictionary lookup words correction; sentiment word recognition; sentential sentiment classification; and features evaluation. Their results showed that they achieved around 63% F1-measure at sentence level.

(Ahmad and Almas, 2005) analyzed financial text by selecting their own sentiment words in Arabic and creating a rule for classifying the stem word when using it in combination with various affixes.

### **Purposes of sentiment analysis**

The purpose of sentiment analysis is to identify opinions or attitude in terms of polarity. It can be used in various fields, such as business, politics and psychology. Therefore, the brief details of some sentiment analysis applications are presented in this section.

#### **1. Business**

Sentiment analysis has been used in many business tasks, such as advertising, marketing, production, etc. In terms of advertising, the internet is the best medium through which to promote businesses as it will reach various groups of customers. Sentiment analysis could be used to help ensure that the website's contents fit with the commercial content so that it is not detrimental to the reputation and popularity of the company and/or brand (Jin *et al.*, 2007).

Marketing and production are the main keys for the company and brand that can use sentiment analysis for predicting pricing and demand of the products. For example, (Mishne and Glance, 2006) analyzed sentiment in weblogs towards movies, both before and after their release, and tested that sentiment is associated with the number of references in the weblogs, which is fewer than that of the box office. The results showed that sentiment can be used to predict ticket sales for a movie, along with other factors such as genre and season.

Moreover, sentiment analysis can be used to analyze product reviews from customers. For example, (Grabner *et al.*, 2012) used sentiment analysis to classify customers' reviews of hotels by using a star rating to categorize the reviews as bad, neutral and good. This task showed that reviews could be classified correctly, probably with 90% accuracy, by using sentiment analysis.

#### **2. Politics**

Various political organizations use sentiment analysis to analyze public opinion in relation to policies, legislation, politics, government agencies, etc. For political postings on microblogs, Twitter has been analyzed by various researchers. For example, (Tumasjan *et al.*, 2010) use more than 100,000 tweets posted in the weeks leading up to the German federal election to predict electronically the outcome. They compare the results with the actual electoral votes. The results showed that the mean absolute error (MAE) of the prediction is only 1.65%. Therefore, it could be said that tweets are sufficiently reliable to predict the outcomes of electronic results. More details of MAE can be found in (Jain and Jain, 1981).

#### **3. Psychology**

The researches in psychology are also concerned with emotion, which plays an important role in dreams (Hobson *et al.*, 1998; Domhoff, 2003; St-Onge *et al.*, 2005). Normally, the emotions in dreams are assessed and analysed by the dreamers themselves. In 2006, sentiment analysis was used to classify structures of dreams' contents, whether they are positive or negative (Nadeau *et al.*, 2006). They used humans to annotate the contents of dream according to four levels. Next, they compared the results with machine learning, which yielded an accuracy rate of 50%, with 0.577 of the mean squared error (MSE). More details of MSE can be found in (Koga *et al.*, 1981).

### **Levels of sentiment analysis**

Sentiment analysis can be performed at various levels: word, phrase, sentence and document. The brief details of each can be found in the following section.

#### **1. Document-level sentiment analysis**

Document-level analysis determines the sentiment of the whole document; for example, news, reviews, forums and blogs. Various machine learning algorithms approach for document level. (Turney, 2002) used unsupervised learning to classify more than 400 reviews. Three steps were used to process the documents. First, they extracted the adjectives and adverbs by using a method of part-of-speech tagger, adopted from (Brill, 1994). Second, Pointwise Mutual Information and Information Retrieval algorithm (PMI-IR) was used to evaluate the sentiment orientation of extracted phrases. Finally, the average

semantic orientation of phrases was calculated and customer reviews were classified as 'recommended' or 'not recommended' by achieving 74.39% accuracy. More details of PMI-IR can be found in (Turney, 2001). (Esuli and Sebastiani, 2005) used semi-supervised learning to determine the orientation of subjective terms. (Pang *et al.*, 2002) used three machine learning algorithms based on supervised learning to classify reviews, whether they are positive or negative.

#### 2. Sentence-level sentiment analysis

There are two tasks at the sentence level. First, the sentences will be classified as subjective or objective. Second, polarity of subjective sentences will be classified. (Yu and Hatzivassiloglou, 2003) developed techniques based on supervised learning to classify sentence level. In their task, the polarity of each subjective sentence was identified by adopting the method from (Turney, 2002); however, it used seed words from (Hatzivassiloglou and McKeown, 1997a) and a statistic algorithm called, 'log-likelihood ratio' to calculate polarity scores. (Pang and Lee, 2004) used minimum cuts in a sentences graph to classify subjective sentences. (Meena and Prabhakar, 2007) classify each sentence in the review by using machine learning to analyze the polarity of phrases and merge them by incorporating the effects of conjunctions to make a decision on the overall polarity of a sentence.

#### 3. Phrase-level sentiment analysis

This sub-section involves the classification of the polarity of phrases, such as noun phrase, verb phrase, prepositional phrase, etc. (Wilson *et al.*, 2005b) used machine learning and a variety of features to classify content polarity at phrase level. First, they analyzed each phrase, whether they were neutral or polar. Next, polar phrases were used and their contextual polarity classified as positive, negative, neutral or both positive and negative using polarity shifters. (Takamura *et al.*, 2007) adopted statistical mechanics called, 'Potts model' to extract the semantic orientations of noun and adjective phrases. (Agarwal *et al.*, 2009) used lexical scores from the Dictionary of Affect in Language (DAL) and syntactic n-grams to predict the polarity of phrases within the sentences.

#### 4. Word-level sentiment analysis

Most tasks use word level to classify at the sentence and document level. Word level is concerned with analysing the polarity of words. There are two methods that can be used to classify sentiment at word level: lexicon-based and corpus-based (Taboada *et al.*, 2009; Wan, 2009; Petz *et al.*, 2012).

- Lexicon-based methods

Measuring the polarity derived from text based on sentiment analysis is involved in these methods (Wan, 2009). Lexicon-based methods can be referred to as dictionary-based methods. Sentiment lexicons are words that have a polarity score (Liu, 2012a). For example, 'good' positive score is 0.75, negative score is 0 and neutral score is 0.25 (Baccianella *et al.*, 2010a). (Kim and Hovy, 2004) assigned a polarity score to a list of words to classify opinion based on the given topic and a set of related text. (Devitt and Ahmad, 2007) explore the calculation of positive or negative in financial news messages. (Wu *et al.*, 2009) assigned polarity scores to the documents to calculate and classify their labels based on a graph-ranking algorithm. (Amiri and Chua, 2012) studied the benefit of sense-level polarity information for the task of sentiment classification.

- Corpus-based methods

These methods concerned train sentiment classification by using corpora of documents that are labelled with polarity (Wan, 2009). The polarity of sentiment did not have to be 'positive' and 'negative'. Moreover, there can be more than two labels of polarity (Read, 2009). (Mihalcea and Liu, 2006) classify the corpus of blog posts from the LiveJournal using labels of 'happy' and 'sad'. (Yerva *et al.*, 2010) classified tweets using sentiment analysis, according to whether or not they are related to a company. (McDonald *et al.*, 2007) investigated predicting sentiment at different levels of granularity for a text using a global structured model. (Keshtkar and Inkpen, 2010) investigate using sentiment analysis to classify paraphrases into various categories. (Grabner *et al.*, 2012) used three labels to classify customers reviews: 'bad', 'neutral' and 'good'. (Pestian *et al.*, 2012) used sentiment classification to analyse emotions in suicide notes.

### Polarity measurement and label classification

This section presents some processes that can be used to measure polarity scores and classify polarity labels.

#### 1. Polarity scores from resources

There are some lexicon resources that consist of polarity scores, such SentiWordNet and SentiStrength. More details of these can be found in section 5. Some researchers adapted those scores for use in their works. For example, (Amiri and Chua, 2012) summed up the values of synsets for each tag on SentiWordNet and assigned labels to them: -1, 0 and +1. Other tags that do not appear in SentiWordNet will assign label '0'. The tags in SentiWordNet are noun, adjective, adverb and verb; for example, the word 'short' has 11 adjective, three noun, seven adverb and two verb senses. According to this, the term 'short' in the adjective tag will be '-1', as the sum of positive scores (0.5) is lower than that of negative scores (3.5) over all 11 adjectives. This is the same for the adverb and verb tags. Meanwhile, the term 'short' in the noun tag has the label '0' because positive and negative scores are zero over three noun senses in SentiWordNet.

#### 2. Human classification

By using humans to classify the contents, the researchers will find more than two annotators to score the words using ranging. Ranging can vary, depending on the agreement between the researchers and annotators. After that, the statistical measure of the agreement of annotators will be used. (Devitt and Ahmad, 2007) used humans to annotate polarity in financial news. They used three annotators to annotate a set of 30 texts ranging from 1 (very negative) to 7 (very positive). Then, Krippendorff's alpha was used to measure the agreement of annotators. More details of this method can be found in (Krippendorff, 1980).

#### 3. Reviews rating

Reviews rating is used in various organizations, such as hotels, cinemas

and restaurants; whereby customers can review their products or/and services. Some researchers used the rating scales to annotate the score of the contents. (Grabner *et al.*, 2012) used sentiment analysis to classify customers' reviews of hotels. They assigned weight to the star rating used to annotate the reviews; for example, 1 star, 2 star, 3 star, 4 star and 5 star are weighted as -2, -1, 0, +1 and +2, respectively. Then, the reviews with values of -2, 0 and +2 are assigned labels as bad, neutral and good, respectively for use in the comparison.

#### 4. Emoticons

The icons that can be used to express emotion are called 'Emoticons' (Witmer and Katzman, 1997; Danet *et al.*, 1997). These are normally used in social networks, such as Facebook and Twitter. For example, (Aisopos *et al.*, 2011) divided tweets in to three groups by using emoticons for classification. If tweets contain positive emoticons, they will be classified as positive and vice versa. Other tweets that did not have positive/negative emoticons will be classified as neutral. However, tweets that contain both positive and negative emoticons are ignored in their study. Their task focused on analyzing the contents of social media by using n-gram graphs, and the results showed that n-grams yielded high accuracy when tested with C4.5 but low accuracy with Naïve Bayes Multinomial (NBM). Both C4.5 and NBM are used for text classification.

#### 5. Feature-based analysis

Feature-based analysis is focused on target entities and components of the opinions. The targets could be service, product, organization, topic, etc. Components can be referred to as attributes and features. (Hu and Liu, 2004) studied customer reviews by focusing on the product features. First, they identified a product's features from the customer's reviews. Next, they identified reviews of each feature, whether they are positive or negative. Finally, they summarized the overall

reviews of each feature and used them in their experiment.

### Resources of sentiment analysis

There are some sentiment analysis resources that can be used to classify contents, such as SentiWordNet, SentiStrength, etc.

#### 1. SentiWordNet

SentiWordNet (Baccianella *et al.*, 2010a) is a freely—available and widely used electronic resource. For example, (Denecke, 2008) used SentiWordNet to determine the polarity of text within a multilingual framework. (Ohana and Tierney, 2009) used SentiWordNet to calculate positive and negative scores to determine sentiment orientation. (Kim and Calvo, 2011) used SentiWordNet as a linguistic lexical resource for sentiment summarization of feedback in academic essay writing. In 2010, the latest version of SentiWordNet was presented to the public (Baccianella *et al.*, 2010b).

SentiWordNet is the result of the automatic annotation of all the synsets of WordNet (Fellbaum, 1998; Princeton University, 2010), according to the notions of positive, negative and neutrality, to which each synset allocates three numerical scores Pos(s), Neg(s) and Obj(s). Each of the three scores ranges from 0.0 to 1.0 and their sum is 1.0 for each synset. This means that there is the possibility of having non-zero scores for all three.

The methods used to generating SentiWordNet were adapted from the methods of PN-polarity and SO-polarity (Esuli and Sebastiani, 2006b). PN-polarity is used to determine whether the opinion is positive or negative, while SO-polarity determines whether the opinion is subject or objective. The methods relies on the quantitative analysis of annotates associated with synsets and on the use of the resulting quantity term representations for semi-supervised synset classification (Esuli and Sebastiani, 2007). Semi-supervised

classification is a machine-learning technique for use with both labelled and unlabelled data. More details of semi-supervised classification can be found in (Zhu *et al.*, 2009b).

#### 2. SentiStrength

SentiStrength (Thelwall *et al.*, 2010b) is also available to use free of charge and has been used by some researchers. For example, (Pfitzner *et al.*, 2012) use SentiStrength to classify sentiment expressed in microblogs. (Preethi *et al.*, 2012) investigated online hotspot forums, using SentiStrength to calculate sentiment scores of the existing text in each forum.

SentiStrength is the sentiment analysis methodology used to judge whether a sentence has a positive or negative sentiment. The methodology was developed using nearly 4,000 comments on MySpace by (Thelwall *et al.*, 2010a). They used three annotators and Krippendorff's alpha to measure their agreement. The data has been separated into two groups: trail data and testing data. Trail data was used to identify algorithms for judgment and suitable scales. Algorithms were identified, ranging from 1 to 5. They were used alongside testing data for final judgment and these will be SentiStrength's lexicon.

### Conclusion

In this paper, the basic terms of sentiment analysis have been described. The goal of sentiment analysis is to determine the polarity of words, phrases, sentences and documents. Sentiment analysis is used in various fields, such as business, politics and psychology. Levels of sentiment analysis and the processes used to generate polarity and labels have been analyzed. SentiWordNet and SentiStrength have been identified as the resources of sentiment analysis. For future work, we plan to investigate machine leaning and other techniques that could be used to



classify the data (for example, FrameNet).

## References

- [1] C. J. Leuba, *Man: A General Psychology*: Holt, Rinehart and Winston, 1961.
- [2] W. K. Richmond, *Teachers and machines: an introduction to the theory and practice of programmed learning*: Collins, 1965.
- [3] A. Hogenboom, F. Boon, and F. Frasincar, "A Statistical Approach to Star Rating Classification of Sentiment," in *Management Intelligent Systems*. vol. 171, J. Casillas, F. J. Martínez-López, and J. M. Corchado Rodríguez, Eds., ed: Springer Berlin Heidelberg, 2012, pp. 251-260.
- [4] M. A. Shaikh, H. Prendinger, and I. Mitsuru, "Assessing Sentiment of Text by Semantic Dependency and Contextual Valence Analysis," presented at the Proceedings of the 2nd international conference on Affective Computing and Intelligent Interaction, Lisbon, Portugal, 2007.
- [5] J. Karlgren, M. Sahlgren, F. Olsson, F. Espinoza, and O. Hamfors, "Usefulness of Sentiment Analysis," in *Advances in Information Retrieval*. vol. 7224, R. Baeza-Yates, A. Vries, H. Zaragoza, B. B. Cambazoglu, V. Murdock, R. Lempel, and F. Silvestri, Eds., ed: Springer Berlin Heidelberg, 2012, pp. 426-435.
- [6] B. Pang and L. Lee, "Opinion Mining and Sentiment Analysis," *Found. Trends Inf. Retr.*, vol. 2, pp. 1-135, 2008.
- [7] U. Waltinger, "German Polarity Clues: A Lexical Resource for German Sentiment Analysis," in *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC)*, 2010, pp. 1638-1642.
- [8] C. J. V. Rijsbergen, *Information Retrieval*: Butterworth-Heinemann, 1979.
- [9] C. Haruechaiyasak, A. Kongthon, P. Palingoon, and C. Sangkeettrakarn, "Constructing Thai Opinion Mining Resource: A Case Study on Hotel Reviews," presented at the Proceedings of the Eighth Workshop on Asian Language Resources, Beijing, China, 2010.
- [10] A. Kongthon, C. Haruechaiyasak, C. Sangkeettrakarn, P. Palingoon, and W. Wunnasri, "HotelOpinion: An opinion mining system on hotel reviews in Thailand," in *Technology Management in the Energy Smart World (PICMET)*, 2011 Proceedings of PICMET '11:, 2011, pp. 1-6.
- [11] X. Wan, "Using bilingual knowledge and ensemble techniques for unsupervised Chinese sentiment analysis," presented at the Proceedings of the Conference on Empirical Methods in Natural Language Processing, Honolulu, Hawaii, 2008.
- [12] L.-W. Ku, T.-H. Huang, and H.-H. Chen, "Using morphological and syntactic structures for Chinese opinion analysis," presented at the Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 3 - Volume 3, Singapore, 2009.
- [13] H. Ghorbel and D. Jacot, "Sentiment analysis of French movie reviews," *Advances in Distributed Agent-Based Retrieval Tools*, pp. 97-108, 2011.
- [14] B. T. Kieu and S. B. Pham, "Sentiment Analysis for Vietnamese," presented at the Knowledge and Systems Engineering (KSE), 2010 Second International Conference on, 2010.

- [15] H. Cunningham, D. Maynard, and K. Bontcheva, Text Processing with Gate (Version 6): Gate, 2011.
- [16] D. Day, C. McHenry, R. Kozierok, and L. Riek, "Callisto: A configurable annotation workbench," in International Conference on Language Resources and Evaluation, 2004.
- [17] D. Thakker, T. Osman, and P. Lakin, "GATE JAPE Grammar Tutorial Version 1.0," 2009.
- [18] K. Ahmad and Y. Almas, "Visualising sentiments in financial texts?," in Information Visualisation, 2005. Proceedings. Ninth International Conference on, 2005, pp. 363-368.
- [19] X. Jin, Y. Li, T. Mah, and J. Tong, "Sensitive webpage classification for content advertising," presented at the Proceedings of the 1st international workshop on Data mining and audience intelligence for advertising, San Jose, California, 2007.
- [20] G. Mishne and N. Glance, "Predicting Movie Sales from Blogger Sentiment," presented at the AAAI 2006 Spring Symposium on Computational Approaches to Analysing Weblogs, Palo Alto, California, USA, 2006.
- [21] D. Grabner, M. Zanker, G. Fliedl, and M. Fuchs, "Classification of customer reviews based on sentiment analysis," presented at the 19th Conference on Information and Communication Technologies in Tourism (ENTER), Helsingborg, Sweden, 2012.
- [22] A. Tumasjan, T. O. Sprenger, P. G. Sandner, and I. M. Welpe, "Predicting elections with twitter: What 140 characters reveal about political sentiment," in Proceedings of the Fourth International AAAI Conference on Weblogs and Social Media, 2010, pp. 178-185.
- [23] J. Jain and A. Jain, "Displacement Measurement and Its Application in Interframe Image Coding," Communications, IEEE Transactions on, vol. 29, pp. 1799-1808, 1981.
- [24] J. A. Hobson, R. Stickgold, and E. F. Pace-Schott, "The neuropsychology of REM sleep dreaming," NeuroReport, vol. 9, pp. R1-R14, 1998.
- [25] G. W. Domhoff, The Scientific Study of Dreams: Neural Networks, Cognitive Development, and Content Analysis: Amer Psychological Assn, 2003.
- [26] M. St-Onge, M. Lortie-Lussier, P. Mercier, J. Grenier, and J. D. Koninck, "Emotions in the Diary and REM Dreams of Young and Late Adulthood Women and Their Relation to Life Satisfaction," Dreaming, vol. 15, pp. 116-128, 2005.
- [27] D. Nadeau, C. Sabourin, J. De Koninck, S. Matwin, and P. Turney, "Automatic dream sentiment analysis," presented at the In: Proceedings of the Workshop on Computational Aesthetics at the Twenty-First National Conference on Artificial Intelligence, Boston, Massachussets, USA, 2006.
- [28] T. Koga, K. Iinuma, A. Hirano, Y. Iijima, and T. Ishiguro, "Motion compensated interframe coding for video conferencing " in Proceedings of national Telecommunications conference, New Orleans, LA, 1981, pp. G5.3.1-G5.3.5.
- [29] P. D. Turney, "Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews," presented at the Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, Philadelphia, Pennsylvania, 2002.

- [30] E. Brill, "Some advances in transformation-based part of speech tagging," presented at the Proceedings of the twelfth national conference on Artificial intelligence (vol. 1), Seattle, Washington, USA, 1994.
- [31] P. D. Turney, "Mining the Web for Synonyms: PMI-IR versus LSA on TOEFL," presented at the Proceedings of the 12th European Conference on Machine Learning, 2001.
- [32] A. Esuli and F. Sebastiani, "Determining the Semantic Orientation of Terms through Gloss Classification," presented at the Proceedings of the 14th ACM international conference on Information and knowledge management, Bremen, Germany, 2005.
- [33] B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up?: sentiment classification using machine learning techniques," presented at the Proceedings of the ACL-02 conference on Empirical methods in natural language processing - Volume 10, 2002.
- [34] H. Yu and V. Hatzivassiloglou, "Towards answering opinion questions: separating facts from opinions and identifying the polarity of opinion sentences," presented at the Proceedings of the 2003 conference on Empirical methods in natural language processing, 2003.
- [35] V. Hatzivassiloglou and K. R. McKeown, "Predicting the semantic orientation of adjectives," presented at the Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics and Eighth Conference of the European Chapter of the Association for Computational Linguistics, Madrid, Spain, 1997.
- [36] B. Pang and L. Lee, "A sentimental education: sentiment analysis using subjectivity summarization based on minimum cuts," presented at the Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics, Barcelona, Spain, 2004.
- [37] A. Meena and T. V. Prabhakar, "Sentence Level Sentiment Analysis in the Presence of Conjuncts Using Linguistic Analysis," in *Advances in Information Retrieval*, vol. 4425, G. Amati, C. Carpineto, and G. Romano, Eds., ed: Springer Berlin Heidelberg, 2007, pp. 573-580.
- [38] T. Wilson, J. Wiebe, and P. Hoffmann, "Recognizing contextual polarity in phrase-level sentiment analysis," presented at the Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing, Vancouver, British Columbia, Canada, 2005.
- [39] H. Takamura, T. Inui, and M. Okumura, "Extracting Semantic Orientations of Phrases from Dictionary," in *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, Rochester, NY, 2007, pp. 292-299.
- [40] A. Agarwal, F. Biadisy, and K. R. McKeown, "Contextual phrase-level polarity analysis using lexical affect scoring and syntactic N-grams," presented at the Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics, Athens, Greece, 2009.
- [41] M. Taboada, J. Brooke, and M. Stede, "Genre-based paragraph classification for sentiment analysis," presented at the Proceedings of the SIGDIAL 2009 Conference: The 10th Annual Meeting of the Special Interest Group on Discourse and Dialogue, London, United Kingdom, 2009.
- [42] X. Wan, "Co-training for cross-lingual sentiment classification," presented at the Proceedings of the Joint

Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1 - Volume 1, Suntec, Singapore, 2009.

[43] G. Petz, M. Karpowicz, H. Furschub, A. Auinger, S. M. Winkler, S. Schaller, and A. Holzinger, "On Text Preprocessing for Opinion Mining Outside of Laboratory Environments," in *Active Media Technology*, R. Huang, A. Ghorbani, G. Pasi, T. Yamaguchi, N. Yen, and B. Jin, Eds., ed: Springer Berlin Heidelberg, 2012, pp. 618-629.

[44] B. Liu, "Chapter 9 Opinion Search and Retrieval," in *Sentiment Analysis and Opinion Mining*, ed: Morgan & Claypool, 2012, p. 108.

[45] S. Baccianella, A. Esuli, and F. Sebastiani, "SentiWordNet," ed, 2010.

[46] S.-M. Kim and E. Hovy, "Determining the sentiment of opinions," presented at the Proceedings of the 20th international conference on Computational Linguistics, Geneva, Switzerland, 2004.

[47] A. Devitt and K. Ahmad, "Sentiment Polarity Identification in Financial News: A Cohesion-based Approach," in *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, 2007, pp. 984-991.

[48] Q. Wu, S. Tan, and X. Cheng, "Graph ranking for sentiment transfer," presented at the Proceedings of the ACL-IJCNLP 2009 Conference Short Papers, Suntec, Singapore, 2009.

[49] H. Amiri and T.-S. Chua, "Sentiment Classification Using the Meaning of Words," in *Intelligent Techniques for Web Personalization and Recommender Systems: AAAI Technical Report WS-12-09*, D. Jannach, S. S. Anand, B. Mobasher, and A. Kobsa, Eds., ed Palo Alto, California: The AAAI Press, 2012.

[50] J. Read, "Weakly Supervised Techniques for the Analysis of Evaluation in Text," PhD Thesis, University of Sussex, 2009.

[51] R. Mihalcea and H. Liu, "A Corpus-based Approach to Finding Happiness," presented at the Proceedings of the AAAI Spring Symposium on Computational Approaches to Weblogs, 2006.

[52] S. R. Yerva, Z. Miklos, and K. Aberer, "It was easy, when apples and blackberries were only fruits," presented at the Third Web People Search Evaluation Forum (WePS-3), CLEF, 2010.

[53] R. McDonald, K. Hannan, T. Neylon, M. Wells, and J. Reynar, "Structured Models for Fine-to-Coarse Sentiment Analysis," presented at the 45th Annual Meeting of the Association of Computational Linguistics, Prague, Czech Republic, 2007.

[54] F. Keshtkar and D. Inkpen, "A corpus-based method for extracting paraphrases of emotion terms," presented at the Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text, Los Angeles, California, 2010.

[55] J. P. Pestian, P. Matykiewicz, M. Linn-Gust, B. South, O. Uzuner, J. Wiebe, K. B. Cohen, J. Hurdle, and C. Brew, "Sentiment Analysis of Suicide Notes: A Shared Task," *Biomed Inform Insights*, vol. 5, pp. 3-16, 2012.

[56] K. H. Krippendorff, *Content analysis: an introduction to its methodology*. Beverly Hills, CA: Calif Sage Publications, 1980.

[57] D. F. Witmer and S. L. Katzman, "Smile when you say that: graphic accents as gender markers in computer-mediated communication," in *Network and Netplay*, S. Fay, M.

Margaret, and R. Sheizaf, Eds., ed: MIT Press, 1997, pp. 3-11.

[58] B. Danet, L. Ruedenberg-Wright, and Y. Rosenbaum-Tamari, "'HMMM...WHERE'S THAT SMOKE COMING FROM?'," *Journal of Computer-Mediated Communication*, vol. 2, 1997.

[59] F. Aisopos, G. Papadakis, and T. Varvarigou, "Sentiment analysis of social media content using N-Gram graphs," presented at the Proceedings of the 3rd ACM SIGMM international workshop on Social media, Scottsdale, Arizona, USA, 2011.

[60] A. McCallum and K. Nigam, "A comparison of event models for Naive Bayes text classification," in *IN AAAI-98 WORKSHOP ON LEARNING FOR TEXT CATEGORIZATION*, 1998, pp. 41-48.

[61] M. Hu and B. Liu, "Mining and summarizing customer reviews," presented at the Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining, Seattle, WA, USA, 2004.

[62] K. Denecke, "Using SentiWordNet for multilingual sentiment analysis," presented at the Data Engineering Workshop, 2008. ICDEW 2008. IEEE 24th International Conference, 2008.

[63] B. Ohana and B. Tierney, "Sentiment classification of reviews using SentiWordNet," presented at the 9th. IT&T Conference, Dublin Institute of Technology, Dublin, Ireland, 2009.

[64] S. M. Kim and R. A. Calvo, "Sentiment-Oriented Summarisation of Peer Reviews." vol. 6738, ed: Springer Berlin / Heidelberg, 2011, pp. 491-493.

[65] S. Baccianella, A. Esuli, and F. Sebastiani, "SentiWordNet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion

Mining," presented at the Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10), Valletta, Malta, 2010.

[66] C. Fellbaum, *WordNet: An Electronic Lexical Database*: Mit Press, 1998.

[67] Princeton University, "WordNet," ed: Princeton University, 2010.

[68] A. Esuli and F. Sebastiani, "SENTIWORDNET: A Publicly Available Lexical Resource for Opinion Mining," presented at the In Proceedings of the 5th Conference on Language Resources and Evaluation (LREC'06), Genoa, Italy, 2006.

[69] A. Esuli and F. Sebastiani, "SENTIWORDNET: A high-coverage lexical resource for opinion mining," *Evaluation*, pp. 1-26, 2007.

[70] X. Zhu, A. B. Goldberg, R. Brachman, and T. Dietterich, *Introduction to Semi-Supervised Learning*: Morgan and Claypool Publishers, 2009.

[71] M. Thelwall, K. Buckley, G. Paltoglou, D. Cai, and A. Kappas, "SentiStrength," ed: University of Wolverhampton, 2010.

[72] R. Pfitzner, A. Garas, and F. Schweitzer, "Emotional Divergence Influences Information Spreading in Twitter," presented at the The 6th International AAAI Conference on Weblogs and Social Media, Dublin, Ireland, 2012.

[73] T. Preethi, K. N. Devi, and V. M. Bhaskaran, "A semantic enhanced approach for online hotspot forums detection," in *International Conference on Recent Trends In Information Technology (ICRTIT 2012)*, 2012, pp. 497-501.

- [74] M. Thelwall, K. Buckley, G. Paltoglou, and D. Cai, "Sentiment strength detection in short informal text," *Journal of the American Society for Information Science and Technology*, vol. 61, pp. 2544-2558, 2010.

Publication 6: The International Workshop on Semantic Evaluation (SemEval 2014)

## **TJP: Identifying the Polarity of Tweets from Context**

**Tawunrat Chalothorn**

Department of Computer Science and  
Digital Technologies  
University of Northumbria at Newcastle  
Pandon Building, Camden Street  
Newcastle Upon Tyne, NE2 1XE, UK  
Tawunrat.chalothorn@northumbria.ac.uk

**Jeremy Ellman**

Department of Computer Science and  
Digital Technologies  
University of Northumbria at Newcastle  
Pandon Building, Camden Street  
Newcastle Upon Tyne, NE2 1XE, UK  
Jeremy.ellman@northumbria.ac.uk

### **Abstract**

The TJP system is presented, which participated in SemEval 2014 Task 9, Part A: Contextual Polarity Disambiguation. Our system is ‘constrained’, using only data provided by the organizers. The goal of this task is to identify whether marking contexts are positive, negative or neutral. Our system uses a support vector machine, with extensive pre-processing and achieved an overall F-score of 81.96%.

### **1 Introduction**

The aim of sentiment analysis is to identify whether the subject of a text is intended to be viewed positively or negatively by a reader. Such emotions are sometimes hidden in long sentences and are difficult to identify. Consequently sentiment analysis is an active research area in natural language processing.

Sentiment is currently conceived in terms of polarity. This has numerous interesting applications. For example, Grabner et al. (2012) used sentiment analysis to classify customers’ reviews of hotels by using a star rating to categorize the reviews as bad, neutral and good. Similarly, Tumasjan et al. (2010) tried to predict the outcome of the German federal election through the analysis of more than 100,000 tweets posted in the lead up. Sentiment analysis has also been used to classify

whether tweets are positive or negative (Nadeau et al. 2006).

This paper presents the TJP system which was submitted to SemEval 2014 Task 9, Part A: Contextual Polarity Disambiguation (Rosenthal et al., 2014). TJP focused on the ‘Constrained’ task.

The ‘Constrained’ task only uses data provided by the organizers. That is, external resources such as sentiment inventories (e.g. Sentiwordnet (Esuli, and Sebastiani 2006)) are excluded. The objective of the TJP system was to use the results for comparison with our previous experiment (Chalothorn and Ellman, 2013). More details of these can be found in section 5.

The TJP system was implemented using a support vector machine (SVM, e.g. Joachims, 1999) with the addition of extensive pre-processing such as stopword removal, negation, slang, contraction, and emoticon expansions.

The remainder of this paper is constructed as follows: firstly, related work is discussed in section 2; the methodology, the experiment and results are presented in sections 3 and 4, respectively. Finally a discussion and future work are given in section 5.

### **2 Related Work**

Twitter is a popular social networking and microblogging site that allows users

to post messages of up to 140 characters; known as 'Tweets'. Tweets are extremely attractive to the marketing sector, since tweets may be searched in real-time. This means marketing can find customer sentiment (both positive and negative) far more quickly than through the use of web pages or traditional media. Consequently analyzing the sentiment of tweets is currently active research task.

The word 'emoticon' is a neologistic contraction of 'emotional icon'. It refers specifically to the use of combinations of punctuation characters to indicate sentiment in a text. Well known emoticons include :) to represent a happy face, and :( a sad one. Emoticons allow writers to augment the impact of limited texts (such as in SMS messages or tweets) using few characters.

Read (2005) used emoticons from a training set downloaded from Usenet newsgroups as annotations (positive and negative). Using the machine learning techniques of Naïve Bayes and SVM, Read (2005) achieved up to 61.50 % and 70.10%, accuracy respectively in determining text polarity from the emoticons used.

Go et al. (2009) used distant supervision to classify sentiment of Twitter, similar to Read (2005). Emoticons were used as noisy labels in training data. This allowed the performance of supervised learning (positive and negative) at a distance. Three classifiers were used: Naïve Bayes, Maximum Entropy and SVM. These classifiers were able to obtain more than 81.30%, 80.50% and 82.20%, respectively accuracy on their unigram testing data.

Aramaki et al. (2011) classified contexts on Twitter related to influenza using a SVM. The training data was

annotated with the polarity label by humans, whether they are positive or negative. The contexts will be labelled as positive if the contexts mention the user or someone close to them has the flu, or if they mention a time when they caught the flu. The results demonstrated that they obtained a 0.89 correction ratio for their testing data against a gold standard.

Finally, a well known paper by Bollen and Mao (2011) identified a correlation between the movements of the Dow Jones stock market index, and prevailing sentiment as determined from twitter's live feed. This application has prompted considerable work such as Makrehchi et al (2013) that has attempted to create successful trading strategies from sentiment analysis of tweets.

These work both the wide ranging applications of analysing twitter data, and the importance of Sentiment Analysis. We now move on to look at our approach to SemEval 2014 task 9.

### 3 Methodology

#### 3.1 Corpus

The training and development dataset of SemEval was built using Tweets from more than one thousand pieces of context. The contexts have various features often used in Tweets, such as emoticons, tags, usernames etc. These features were extracted from the datasets before training for the supervised machine learning model.

During initial pre-processing of the datasets, emoticons were labelled by matching with the emoticons that have been collect manually from the dataset. Those labelled were matched against a well-known collection of emoticons<sup>69</sup>.

---

<sup>69</sup>[http://en.wikipedia.org/wiki/List\\_of\\_emoticons](http://en.wikipedia.org/wiki/List_of_emoticons)



Subsequently, negative contractions<sup>70</sup> were expanded in place and converted to full form (e.g. don't -> do not). Moreover, the features of twitters were also removed or replaced by words such as twitter usernames, URLs and hashtags.

A Twitter username is a unique name that shows in the user's profile and may be used for both authentication and identification. This is shown by prefacing the username with an @ symbol. When a tweet is directed at an individual or particular entity this can be shown in the tweet by including @username. For example a tweet directed at 'tawunrat' would include the text @tawunrat. Before URLs are posted in twitter they are shortened automatically to use the t.co domain whose modified URLs are at most 22 characters. However, both features have been removed from the datasets. For the hashtags, they are used for represent keyword and topics in twitter by using # follow by words or phrase such as #newcastleuk. This feature has been replaced with the following word after # symbol. For example, #newcastleuk was replaced by newcastleuk.

Frequently repeated letters are used in tweets for emphasis. These were reduced and replaced using a simple regular expression by two of the same character. For example, happpppppy will be replaced with happy, and coollllll will be replaced by cooll. Next, special character such as [.,{,},?,and ! were also removed. Slang and contracted words were converted to their full form. E.g. 'fyi' was converted to 'for your information'. Finally, NLTK (Bird et al. 2009) stopwords such as 'a',

'the', etc., were removed from the datasets.

### 3.2 Classifier

Our system uses the SVM classifier model (Hearst et al., 1998, Cristianini and Shawe-Taylor, 2000), which is based on SVM-light (Joachims, 1999). SVM is a binary linear classification model with the learning algorithm for classification and regression analyzing the data and recognizing the pattern.

Training SVMLight requires data to be formulated into vectors of attribute value pairs preceded by a numeric value. For example,

```
<target>          <feature>:<value>  <feature>:<value>  ...
<feature>:<value> # <info>
```

Here, 'target' represents the polarity of a sentence or tweet; 'feature' refers to a term in the document, and 'value' refers to a feature weight. This could be used as the relative frequency of a term in the set of documents, or Tf-Idf. Tf-idf is the combination of term frequency (tf) and inverse document frequency (idf), is a weight value often used in text mining and information retrieval. This weight is a statistical measure used to evaluate the relative important of word in a document in the collection (Manning et al., 2008).

$$tf - idf_{t,d} = tf_{t,d} * idf_t \quad (1)$$

where  $tf - idf_{t,d}$  is the weighting the scheme assigns to term  $t$  in document  $d$

Term frequency (tf) is used to measure how frequent the term appears in the document.

$$tf_{t,d} = \frac{n_{t,d}}{\sum_k n_{k,d}} \quad (2)$$

where  $n_{t,d}$  is the number of term  $t$  appears in a document  $d$ .  $\sum_k n_{k,d}$  is the total number of terms  $k$  in the document  $d$ .

<sup>70</sup>[http://en.wikipedia.org/wiki/English\\_auxiliaries\\_and\\_contractions#Negative\\_contractions](http://en.wikipedia.org/wiki/English_auxiliaries_and_contractions#Negative_contractions)

Inverse document frequency (idf) is used to measure how important the term is – i.e. whether the term is common or rare in the collection.

$$idf_t = \log \frac{D}{d_t} \quad (3)$$

where  $D$  is the total number of documents in the collection in corpus.  $d_t$  is the number of documents  $d$  which term  $t$  appears.

Therefore, we chose to work with both of these to observe which yielded the best results in the polarity classification.

The default settings of SVMLight were used throughout. This meant that we used a linear kernel that did not require any parameters.<sup>71</sup>

#### 4 Experiment and Results

In our experiment, we used the datasets and evaluated the system using the F-score measurement. During pre-processing features were extracted from both datasets. First, we used a frequency of word as a featured weight by calculating the frequency of word in the dataset and, during pre-processing, we labelled the emotions in both datasets. The results revealed a lower than average F-score at 34.80%. As this was quite low we disregarded further use of term frequency as a feature weight. We moved on to use Tf-Idf as the feature weight and, again, emoticons in both datasets were labelled. The score of 78.10% was achieved. Then, we kept the pre-processing of the training set stable by combining the features to extract from the testing data. These results are presented in Table 1.

The highest score of 81.96% was recorded when all the features were combined and extracted from both datasets.

The lowest score of 36.48% was recorded when emoticons were extracted from testing data and all features were extracted from training datasets. The

results of the highest scoring experiment were submitted to the task organizers.

Following solution submissions, the task organizers announced the scores by separating the data into the following five groups: LiveJournal2014; SMS2013; Twitter2013; Twitter2014; and Twitter2014 Sarcasm. This would allow the identification of any domain dependent effects. However, the results showed that we achieved above average in all the datasets, as illustrated in Figure 1.

#### 5 Conclusion and Future work

The TJP system participated in SemEval 2014 Task 9, Part A: Contextual Polarity Disambiguation. The system exploited considerable pre-processing, before using the well known, SVMLight machine learning algorithm (Joachims. 1999). The pre-processing used several twitter specific features, such as hashtags and ids, in addition to more traditional Information Retrieval concepts such as the Tf-Idf heuristic (Manning et al., 2008). The results showed that the combination of all features in both datasets achieved the best results, at 81.96%.

An aspect of this contribution is the comparative analysis of feature effectiveness. That is, we attempted to identify which factor(s) made the most significant improvement to system performance. It is clear the pre-processing had a considerable effect on system performance. The use of a different learning algorithm also contributed to performance since, on this task, SVMLight performed better than the Naive Bayes algorithm that was used by our team in 2013.

Sentiment resources was not been used in our system in SemEval 2014 as same as in SemEval 2013 whilst other

---

<sup>71</sup> Based on SVMLight

user groups have employed a variety of resources of different sizes, and accuracy (Wilson et al., 2013). These points lead to the following plan for future activities.

Our future work is to rigorously investigate the success factors for sentiment analysis, especially in the twitter domain. More specifically, we have formulated the following research questions as a result of our participation in SemEval

- Are Sentiment resources essential for the Sentiment Analysis task?
- Can the accuracy and effectiveness of sentiment lexicons be measured? If so,

which feature of the resource (accuracy vs. coverage) is the most effective metric.

- Might it be more effective to use a range of sentiments (e.g. [-1.0 .. 1.0]), rather than binary approach(e.g. positive and negative) taken in SemEval 2013, and 2014?
- Is one machine learning algorithm sufficient, and if so which is it? Or, alternately would an ensemble approach (Rokach, 2005) significantly improve performance?

Testing Training	Emoticon	Negation	@user URL	HashTag	Repeated letters	Special characters	Slang	Stopwords
Emoticon	78.10%	75.18%	75.18%	75.25%	75.25%	76.35%	76.26%	68.19%
Negation	63.56%	79.06%	79.06%	75.25%	79.14%	80.07%	80.00%	69.70%
@user, URL	63.54%	79.05%	79.05%	79.12%	79.14%	80.07%	80.00%	69.70%
HashTag	63.59%	79.08%	79.08%	79.11%	79.18%	80.10%	80.03%	69.67%
Repeated letters	63.60%	79.08%	79.10%	79.14%	79.18%	80.11%	80.02%	69.74%
Special characters	67.87%	79.10%	78.55%	79.17%	78.62%	80.82%	80.69%	69.62%
Slang	68.39%	78.39%	78.39%	78.62%	78.45%	80.70%	80.85%	69.56%
Stopwords	36.48%	64.67%	64.67%	78.45%	64.67%	64.82%	64.82%	81.96%

Table 1: The results of each feature analyzed in the approach of TF-IDF<sup>72</sup>

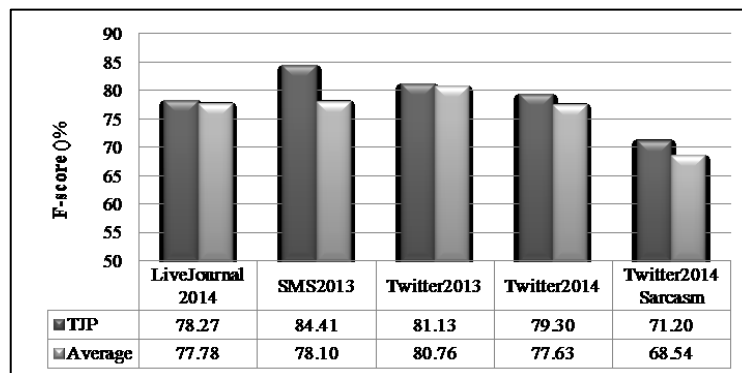


Figure 1: The comparison of TJP and average scores

<sup>72</sup> The results in the table are from the test set 2014 in task 2A.

## References

- Alec Go, Richa Bhayani and Lei Huang. 2009. Twitter sentiment classification using distant supervision. CS224N Project Report, Stanford, 1-12.
- Andrea Esuli, Fabrizio Sebastiani 2006 "SENTIWORDNET: A Publicly Available Lexical Resource for Opinion Mining" in Proceedings of the 5th Conference on Language Resources and Evaluation, LREC (2006), pp. 417-422
- Andranik Tumasjan, Timm O. Sprenger, Philipp G. Sandner and Isabell M. Welpe. 2010. "Predicting elections with twitter: What 140 characters reveal about political sentiment," in Proceedings of the Fourth International AAAI Conference on Weblogs and Social Media, pp. 178-185.
- Christopher D. Manning, Prabhakar Raghavan and Hinrich Schütze, Introduction to Information Retrieval, Cambridge University Press. 2008. ISBN: 0521865719.
- David Nadeau, Catherine Sabourin, Joseph De Koninck, Stan Matwin and Peter D. Turney. 2006. "Automatic dream sentiment analysis," presented at the In: Proceedings of the Workshop on Computational Aesthetics at the Twenty-First National Conference on Artificial Intelligence, Boston, Massachusetts, USA.
- Dietmar Grabner, Markus Zanker, Gunther Fliedl and Matthias Fuchs. 2012. "Classification of customer reviews based on sentiment analysis," presented at the 19th Conference on Information and Communication Technologies in Tourism (ENTER), Helsingborg, Sweden.
- Eiji Aramaki, Sachiko Maskawa and Mizuki Morita. 2011. Twitter catches the flu: detecting influenza epidemics using Twitter. Proceedings of the Conference on Empirical Methods in Natural Language Processing. Edinburgh, United Kingdom: Association for Computational Linguistics.
- Johan. Bollen and Huina. Mao. Twitter mood as a stock market predictor. IEEE Computer, 44(10):91–94.
- Jonathon Read. 2005. Using emoticons to reduce dependency in machine learning techniques for sentiment classification. Proceedings of the ACL Student Research Workshop. Ann Arbor, Michigan: Association for Computational Linguistics.
- Lior Rokach. 2005. Chapter 45 Ensemble Methods for Classifiers. In: Oded Maimon and Lior Rokach (eds.) Data Mining and Knowledge Discovery Handbook. Springer US.
- Marti A. Hearst, Susan T. Dumais, Edgar Osman, John Platt and Bernhard Scholkopf . 1998. Support vector machines. IEEE, Intelligent Systems and their Applications, 13, 18-28.
- Masoud Makrehchi, Sameena Shah and Wenhui Liao. 2013. Stock Prediction Using Event-Based Sentiment Analysis. Web Intelligence (WI) and

- Intelligent Agent Technologies (IAT), 2013 IEEE/WIC/ACM International Joint Conferences on., 337-342.
- Nello Cristianini and John Shawe-Taylor. 2000. An introduction to support vector machines and other kernel-based learning methods, Cambridge university press.
- Sara Rosenthal, Preslav Nakov, Alan Ritter and Veselin Stoyanov. 2014. SemEval-2014 Task 9: Sentiment Analysis in Twitter. International Workshop on Semantic Evaluation (SemEval-2014). Dublin, Ireland.
- Steven Bird, Ewan Klein and Edward Loper. 2009. NLTK: Natural language processing with Python, O'Reilly.
- Takeshi Sakaki, Makoto Okazaki and Yutaka Matsuo. 2010. Earthquake shakes Twitter users: real-time event detection by social sensors. Proceedings of the 19th international conference on World wide web. Raleigh, North Carolina, USA: ACM.
- Tawunrat Chalothorn and Jeremy Ellman. 2013. TJP: Using Twitter to Analyze the Polarity of Contexts. Second Joint Conference on Lexical and Computational Semantics (\*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013). Atlanta, Georgia, USA: Association for Computational Linguistics.
- Theresa Wilson, Zornitsa Kozareva, Preslav Nakov, Alan Ritter, Sara Rosenthal and Veselin Stoyanov. 2013. SemEval-2013 Task 2: Sentiment Analysis in Twitter. Proceedings of the 7th International Workshop on Semantic Evaluation. Association for Computational Linguistics.
- Thorsten Joachims. 1999. Making large-scale support vector machine learning practical. Advances in kernel methods. MIT Press.

## Two classifiers in arbiter tree to analyse data

Tawunrat Chalothorn and Jeremy Ellman

Department of Computer Science and Digital Technologies

University of Northumbria at Newcastle, Pandon Building,

Camden Street Newcastle Upon Tyne, NE2 1XE, United Kingdom

{tawunrat.chalothorn, jeremy.ellman}@northumbria.ac.uk

### **Abstract**

*This paper reports on the use of ensemble learning to classify the sentiment of tweets as being either positive or negative. Tweets were chosen because Twitter is both a popular tool and a public, human annotated dataset was made available as part of the SEMVAL 2013 competition. We report on an approach to classification that contrasts single machine learning algorithms with a combination of algorithms in an ensemble learning approach. The single machines learning algorithms used were Support Vector Machine (SVM) and Naïve Bayes (NB) while the method of ensemble learning was the arbiter tree. Our system achieved an F score using the arbiter tree at 83.55% which was the same as SVM but quite slightly than Naïve Bayes algorithm.*

**Keywords:** *Tweets, contexts, positive, negative, natural language processing, ensemble learning*

## **I. INTRODUCTION**

The research area of natural language processing (NLP) is composed of various tasks; one of which is sentiment analysis. The main goal of sentiment analysis is to identify the polarity of natural language text. Sentiment analysis can be referred to as opinion mining; studying opinions, appraisals and emotions towards entities, events and their attributes. Sentiment analysis is a popular research area in NLP that aims to identify opinions or attitudes in terms of polarity. Currently, Twitter is a popular microblogging tool where users are increasing by the minute. Twitter allows users to post messages of up to 140 characters each time. These are called ‘Tweets’, which are often used to convey opinions about different topics. Consequently, various researchers are interested in classifying Tweets by using sentiment analysis.

This paper introduces the novelty of using arbiter tree (Chan and Stolfo, 1993; Chan and Stolfo, 1995; Chan, 1996; Chan and Stolfo, 1997; Prodromidis *et al.*, 2000), to classify the contexts of Tweet datasets and use SMS datasets to evaluate the system. Arbiter tree (Chan and Stolfo, 1993; Chan and Stolfo, 1995; Chan, 1996; Chan and Stolfo, 1997; Prodromidis *et al.*, 2000) has been chosen because it has not yet been used in sentiment analysis to classify Tweets or SMS datasets. The basic idea is to divide the training data into subsets, apply the learning algorithm to each one and merge the resulting inducers. The main task is to find the solution to combining the right learning model in order to achieve better results. Our main contribution is to propose and experiment with a combination of two machine learning, based on the use of the arbiter tree algorithm (Chan and Stolfo, 1993; Chan and Stolfo, 1995; Chan, 1996; Chan and Stolfo, 1997;

Prodromidis *et al.*, 2000). The remainder of this paper is constructed as follows: the detail of the corpus used is discussed in section 2; the methodology with data pre-processing and details of classifier are presented in section 3; section 4 discusses the details of the experiment and results. Finally, a conclusion and recommendations for future work are provided in section 6.

## II. RELATED WORKS

Machine leaning is well-known and widely used in various researches. For example, (Go *et al.*, 2009) used three machine learning algorithms to classify sentiment of Twitter: Naïve Bayes (Liangxiao *et al.*, 2009), Maximum Entropy Modelling (Baldwin, 2009) and Support Vector Machine (Hearst *et al.*, 1998; Cristianini and Shawe-Taylor, 2000). Emoticons have been used as labels (positive and negative) in training data to perform supervised learning. There are two features that were used in the experiment: unigram and part-of-speech. The results from unigram showed that, (Go *et al.*, 2009) achieved 81.3%, 80.5% and 82.2% from three machine learning algorithms, respectively. On the other hand, the results from the combination of unigram and part-of-speech achieved lower accuracy at 79.9%, 79.9% and 81.9% from three machine learning algorithms, respectively. (Go *et al.*, 2009) used single machine leaning algorithm but will the performance achieved better accuracy if used the combination of machine leaning algorithms? This question has not been answered.

(Yerva *et al.*, 2010) used Support Vector Machine (Hearst *et al.*, 1998; Cristianini and Shawe-Taylor, 2000) to classify tweets whether the contexts are related to the company or not. The dataset was obtained from WePS-3. WePS-3 is a workshop that focuses on share tasks on the Searching Information about Entities in the Web. For solving the problem, (Yerva *et al.*, 2010) built corpus by collecting keywords that related to the company by using six profiles. The first profile, keywords that relevant to the company and presented on the company homepage that was provided by WePS-3 was extracted and named as, homepage profile. The second profile, the keywords from meta tags of the webpages were collected and named as, metadata profile. The third profile, (Yerva *et al.*, 2010) used WordNet to find the keywords of the category that the company belong to and named as, category profile. The Forth profile, the keywords that closely related to the company were gotten Google Sets and named as, googleset profile. Google Sets is a source for obtaining common knowledge about the company by identifying and generating the lists of the items that might related to the company such as the companies that similar or competitor or products. In the mid of 2011, Google Sets was discontinued from Google.<sup>73</sup> The fifth and sixth profiles are the collection of the keyword from users' feedback in both positive and negative and named as, positive profile and negative profile, respectively. After getting all profiles, (Yerva *et al.*, 2010) separated the use of these profiles into four tasks: use all profiles, use all profiles except the negative feedback, use all profiles except the category profile and use only home page. The results showed that, the accuracy performance achieved F-score at 59.50%, 62%, 60% and 48%, respectively. In this experiment, Support Vector Machine were used but how much the accuracy could be achieved from using the others machine leaning algorithms? This question has not been answered.

---

<sup>73</sup> <http://googlesystem.blogspot.co.uk/2011/08/google-sets-will-be-shut-down.html>

(Troussas *et al.*, 2013) used three machine learning algorithms: Naïve Bayes (Liangxiao *et al.*, 2009), Rocchio (Salton, 1971) and Perceptron (Rosenblatt, 1957) to classify contents from Facebook by using positive and negative emoticons. Rocchio (Salton, 1971) is not a machine learning but it is text classifier which based on relevance feedback that was introduced by (Salton, 1971). On the other hand, Perceptron (Rosenblatt, 1957) is supervised machine learning with the attempt for finding a hyperplane that separated two sets of point (Rojas, 1996). The datasets were collected by using Facebook API<sup>74</sup>. Facebook API is a platform for building application that available to the Facebook's users. API allow the application to access to the users' information and social connection for connecting to the application for posting the activities or news on users' profile pages of Facebook which subject to the privacy setting of the users (Ortiz, 2010). The results showed that, F-score accuracy achieved at 72%, 74% and 60% for using Naïve Bayes (Liangxiao *et al.*, 2009), Rocchio (Salton, 1971) and Perceptron (Rosenblatt, 1957), respectively. If three machine learning algorithms were combined together, will the accuracy performance achieved better than single machine learning algorithms? This question has not been answered.

### III. CORPUS

The datasets used in our experiment are from SemEval 2013 (Wilson *et al.*, 2013). The data were gathered from Twitter; a well-known and increasingly popular microblogging site. Twitter allows its users to post messages, or 'Tweets', of up to 140 characters each time, which are available for immediate download over the Internet. Tweets are extremely interesting in marketing terms, since their rapid public interaction can either indicate customer success or presage public relations disasters far more quickly than web pages or traditional media. Consequently, the content of tweets and identifying their sentiment polarity as positive or negative is a current active research topic.

The datasets are composed of training data, testing data and gold standard. Gold standard refers to the testing data labelled with the correct polarity. However, these datasets were annotated using five Mechanical Turk workers, also known as Turkers (Wilson *et al.*, 2013). For each sentence, they will mark by using the start and end point of their opinion for the phrase or word, and state whether it is negative, neutral or positive. Then, the words that appear three times from five votes will be assigned the label. In addition to Tweets, SMS messages are used to evaluate the system. SMS messages are also obtained from the organiser of SemEval 2013 (Wilson *et al.*, 2013). Only the datasets labelled as positive and negative will be used in this research.

### IV. METHODOLOGIES

#### 3.1. Data pre-processing

For the process of data pre-processing, emoticons were labelled by matching those that have been collected manually from the dataset against a well-known collection of emoticons. Subsequently, negative contractions were expanded and converted to full form

---

<sup>74</sup> <https://developers.facebook.com/docs/reference/fql>



(e.g. don't -> do not). Moreover, the features of Tweets were removed or replaced by words, such as Twitter usernames, URLs and hashtags.

A Twitter username is a unique name displayed in the user's profile and may be used for both authentication and identification. This is shown by prefacing the username with an @ symbol. When a Tweet is directed at an individual or particular entity, this can be shown in the tweet by including @username. For example, a Tweet directed at 'som' would include the text @som. Before URLs are posted to Twitter, they are shortened automatically to use the t.co domain whose modified URLs are a maximum of 22 characters. However, both features have been removed from the datasets. Hashtags are used to represent keywords and topics in Twitter by using # followed by words or phrases, such as #newcastleuk. This feature has been replaced with the following word after the # symbol. For example, #newcastleuk was replaced by newcastleuk.

Frequently, repeated letters are used to provide emphasis in Tweets. These were reduced and replaced using a simple regular expression by two of the same characters. For example, happpppppy will be replaced with happy, and coollllll will be replaced by cool. Next, special characters were removed, such as [, {, ?, and !. Slang and contracted words were converted to their full form; for example, 'fyi' became 'for your information'. Finally, Natural Language Toolkit (NLTK) (Bird *et al.*, 2009b) stopwords were removed from the datasets, such as 'a', 'the', etc..

Furthermore, three sentiment lexicons were used in this experiment. They are Bing Liu Lexicon (HL) (6780 words), collected over many years by (Hu and Liu, 2004). They began to collect lexicons in 2004, during the course of their work on online customer product reviews (Hu and Liu, 2004). MPQA Subjective Lexicon (MPQA) (8221 words) was created by (Wilson *et al.*, 2005a) using a set of approximately 400 documents. AFINN Lexicon (AFINN) (2477 words) was created from Twitter between 2009-2011 by (Nielsen, 2011a) for use in the United Nation Climate Conference (COP15).

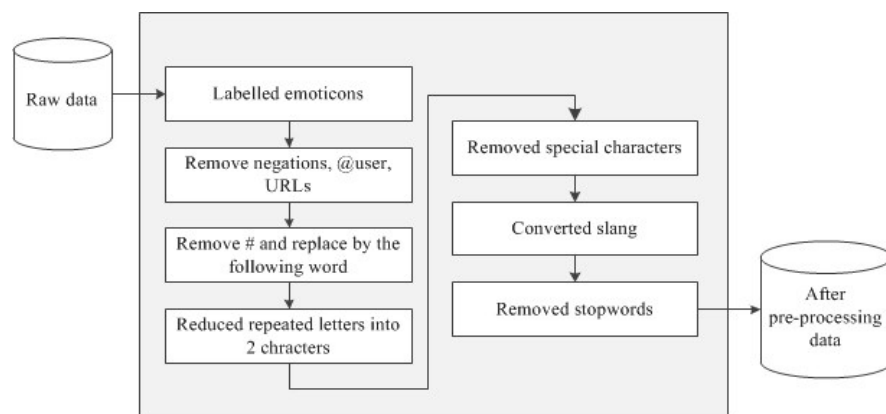


Figure 1: Flowchart of data pre-processing

### 3.2. Arbiter Tree

Arbiter tree (Chan and Stolfo, 1993; Chan and Stolfo, 1995; Chan, 1996; Chan and Stolfo, 1997; Prodromidis *et al.*, 2000) is a method that uses training data output

classified using base classifiers with selection rules. Selection rules are used to compare the prediction of based classifiers for choosing the training dataset for the arbiter. Then, the final prediction is decided according to the base classifiers and arbiter by using arbitration rules with the aim of learning from incorrect classifications (Chan and Stolfo, 1993).

In the process of making the training data for arbiter from (Chan and Stolfo, 1993) mentioned using four training data (T1-4) subsets and four classifiers (C1-4). Next, unite the results T1 and T2, and used selection rules to generate a training set for arbiter A12 with the same learning algorithm used in the initial classifiers. This process is similar to arbiter A34, which used the training data that unite from T3 and T4, and then, the first level of arbiter is produced. After obtaining the results from T12 and T34, they will be united to form a training dataset for the root arbiter A14, as illustrated in Figure 2.

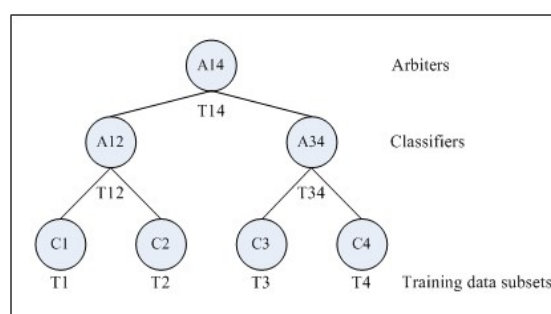


Figure 2: Flowchart to make training dataset for arbiter tree (Chan and Stolfo, 1993)

### 3.3. Support Vector Machine

For using arbiter tree (Chan and Stolfo, 1993; Chan and Stolfo, 1995; Chan, 1996; Chan and Stolfo, 1997; Prodromidis *et al.*, 2000) in our experiment, Support Vector Machine (SVM) (Hearst *et al.*, 1998; Cristianini and Shawe-Taylor, 2000) and Naïve Bayes (NB) (Liangxiao *et al.*, 2009) will be used as classifiers. SVM (Hearst *et al.*, 1998; Cristianini and Shawe-Taylor, 2000) is a binary linear classification model with the learning algorithm for classification and regression analysis of data, and recognising the pattern. The purpose of SVM is to separate datasets into classes and discover the decision boundary (hyper-plane). To find the hyper-plane, the maximum distance between classes (margin) will be used with the closest data points on the margin (support vector). In our research, we used the default setting of SVMLight<sup>75</sup> for the SVM classifier model. SVMLight is an implementation of SVM in C.

### 3.4. Naïve Bayes

Naïve Bayes (NB) algorithm (Liangxiao *et al.*, 2009) is a classification algorithm based on Bayes' theorem that underlies the naïve assumption that attributes within the same case are independent given the class label (Elangovan *et al.*, 2010). This is also known as the state-of-art Bayes rules (Cufoglu *et al.*, 2008). NB (Liangxiao *et al.*, 2009) constructs the

---

<sup>75</sup> <http://svmlight.joachims.org/>

model by adjusting the distribution of the number for each feature. For example, in the text classification, NB regards the documents as a bag-of-words, from which it extracts features. In this research, the NB algorithm was used from the NLTK. NLTK) is a widely-used machine learning, open source, developed using Python and comprising the WordNet interface.

## V. EXPERIMENT AND RESULTS

In our experiment, the idea from (Chan and Stolfo, 1993) has been adapted, as we use only two classifiers with one training data. Therefore, from the flowchart for creating the training data in Figure 2 will be changed to that presented in Figure 3 as only two classifiers.

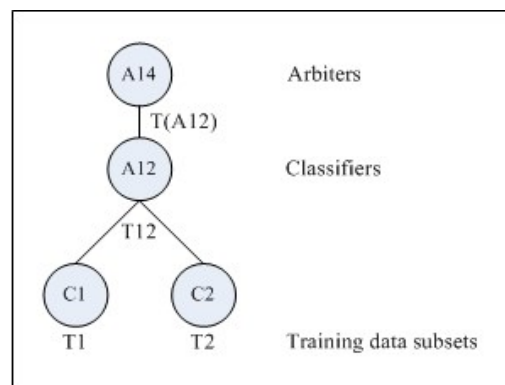


Figure 3: Flowchart to make a training dataset for two classifiers in arbiter tree

In order to build the training data, all selection rules from (Chan and Stolfo, 1993) were adapted and used in this experiment. The processes for creating training data are detailed below:

- i. Base training data was trained into base classifier, which are SVM (Hearst *et al.*, 1998; Cristianini and Shawe-Taylor, 2000) and NB (Liangxiao *et al.*, 2009). The base training data is yielded from the combination of the sentiment lexicons noted in section 3.1. They were combined by removing the words that duplicate, overlap and contradict in sentiment (Melville *et al.*, 2009; Yuan *et al.*, 2013; Refaee and Rieser, 2014; Wang and Cardie, 2014).
- ii. After obtaining the results from the base classifier, they were united and passed into selection rules. There are three versions of selection rules:
  - a. Selection rule 1 is the different results from classifiers 1 and 2
  - b. Selection rule 2 is the union of the results from selection rule 1 and the results from classifiers 1 and 2 that they are the same prediction but incorrect
  - c. Selection rule 3 is the union of selection rules 1 and 2 and the results from classifiers 1 and 2 that they are the same prediction and correct.
- iii. As in the arbiter tree algorithm (Chan and Stolfo, 1993; Chan and Stolfo, 1995; Chan, 1996; Chan and Stolfo, 1997; Prodromidis *et al.*, 2000), (Chan and Stolfo, 1993) did not mention clearly how to use the selection rules; therefore, they will be adapted from the flowchart presented in Figure 3. The data from selection rules 1 and 2 were trained back in base classifiers; then, their results were

combined for processing selection rule 3. This data of selection rule 3 is the final training data for arbiter. The flowchart of these processes is presented in Figure 4.

After obtaining the final training data for arbiter, they were used in the process of final classification for the final prediction results. During this process, the base classifier will be trained by using base training data, while the arbiter is trained by using arbiter training data to classify the test set. Next, their results will go through the process of arbiter rules for the final prediction results. There are two versions of arbiter rules. The first uses the majority vote of prediction from the base classifier and the arbiter prediction. If the results of predictions 1 and 2 are equal, the results from prediction 2 will be used. Conversely, the arbiter results will be used. In the second version, if the results of predictions 1 and 2 are not equal, the different arbiter results will be used. If the results of prediction 1 are equal to those of the correct arbiter, use the correct arbiter results. In contrast, the results from arbiter tree that are incorrect will be used.

The datasets of Tweets and SMS were tested in arbiter tree (Chan and Stolfo, 1993; Chan and Stolfo, 1995; Chan, 1996; Chan and Stolfo, 1997; Prodromidis *et al.*, 2000). Their results are presented in Table 1. Following the comparison between arbiter and base classifier (Table 2), the results of Tweets using arbiter rules version 1 did not make any change and achieved the same F-score as SVM (Hearst *et al.*, 1998; Cristianini and Shawe-Taylor, 2000) at 83.55%; meanwhile, the results from arbiter rules version 2 achieved a better F-score than NB (Liangxiao *et al.*, 2009) at 81.94%, but still lower than SVM (Hearst *et al.*, 1998; Cristianini and Shawe-Taylor, 2000). Conversely, the results of the SMS dataset showed that, the results from arbiter rule version 1 and 2 achieved better F-score than base classifiers at 85.78% and 85.65%, respectively.

Table 1: The results of Tweets and SMS dataset from arbiter tree

	<b>Tweet dataset</b> Avg. F-score (%)	<b>SMS dataset</b> Avg. F-score (%)
Arbiter rules version 1	83.55	85.78
Arbiter rules version 2	81.94	85.65

Table 2: The results of Tweets and SMS dataset from base classifiers

	<b>Tweet dataset</b> Avg. F-score (%)	<b>SMS dataset</b> Avg. F-score (%)
SVM	83.55	85.49
NB	81.54	85.05

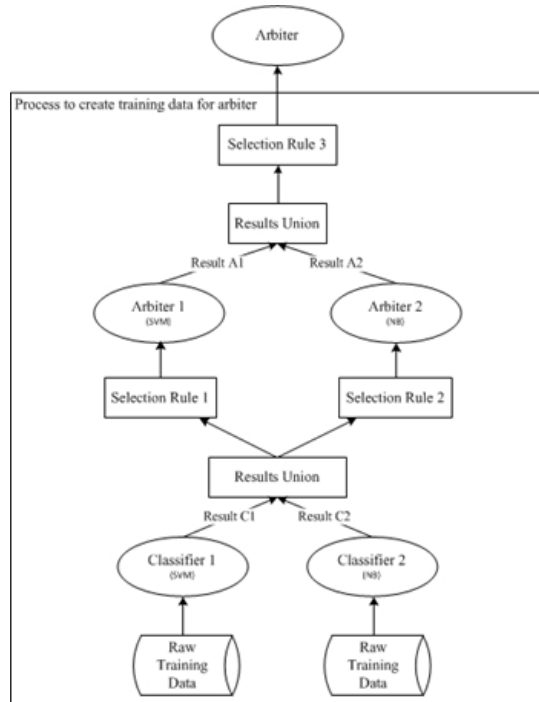


Figure 4: Process for making training data for arbiter

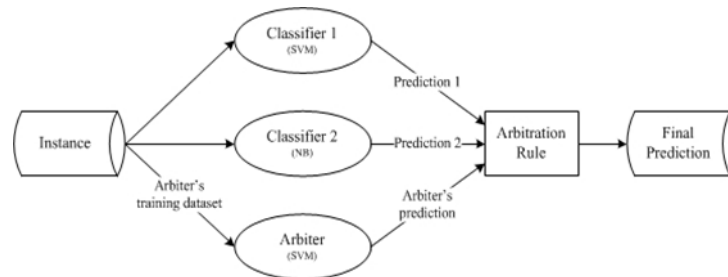


Figure 5: Process for final prediction of the testing data of arbiter tree

## VI. CONCLUSION AND FUTURE WORK

In this experiment, the novelty of using the arbiter tree algorithm (Chan and Stolfo, 1993; Chan and Stolfo, 1995; Chan, 1996; Chan and Stolfo, 1997; Prodromidis *et al.*, 2000) to classify Tweets and SMS datasets has been demonstrated and clearly explained. The use of ensemble learning might not always have achieved the most accuracy; however, the results from the classification of SMS dataset, which we used to evaluate our system, showed that they were able to achieve an F-score of 85.78%, which is better than both base classifiers.

For future work, the sister of arbiter tree (Chan and Stolfo, 1993; Chan and Stolfo, 1995; Chan, 1996; Chan and Stolfo, 1997; Prodromidis *et al.*, 2000), called the combiner tree (Chan and Stolfo, 1997), will be researched in detail and the combination will be studied with the aim of improving the performance accuracy. Combiner tree (Chan and Stolfo, 1997) is a method that is similar to arbiter tree (Chan and Stolfo, 1993; Chan and Stolfo, 1995; Chan, 1996; Chan and Stolfo, 1997; Prodromidis *et al.*, 2000) but is trained directly by the training output from base classifiers that have passed the composition rules. The reason that, arbiter tree (Chan and Stolfo, 1993; Chan and Stolfo, 1995; Chan,

1996; Chan and Stolfo, 1997; Prodromidis *et al.*, 2000) and combiner tree (Chan and Stolfo, 1997) were used, is that the results of them will be used for comparison with the results from stacking (Wolpert, 1992) for analysing which methods of ensemble learning that achieved better approach in the sentiment analysis task of Tweets.

## REFERENCES

- [1] P. K. Chan and S. J. Stolfo, "Toward parallel and distributed learning by meta-learning," in The International Association for the Advancement of Artificial Intelligence (AAAI) workshop in Knowledge Discovery in Databases, 1993, pp. 227-240.
- [2] P. K. Chan and S. J. Stolfo, "Learning Arbiter and Combiner Trees from Partitioned Data for Scaling Machine Learning," in Conference on Knowledge Discovery and Data Mining (KDD), 1995, pp. 39-44.
- [3] P. K. Chan, "An extensible meta-learning approach for scalable and accurate inductive learning," 1996.
- [4] P. K. Chan and S. J. Stolfo, "On the accuracy of meta-learning for scalable data mining," *Journal of Intelligent Information Systems*, vol. 8, pp. 5-28, 1997.
- [5] A. Prodromidis, P. Chan, and S. Stolfo, "Meta-learning in distributed data mining systems: Issues and approaches," *Advances in distributed and parallel knowledge discovery*, vol. 3, 2000.
- [6] A. Go, R. Bhayani, and L. Huang, "Twitter sentiment classification using distant supervision," *CS224N Natural Language Processing*, Project Report, Stanford, pp. 1-12, 2009.
- [7] J. Liangxiao, H. Zhang, and C. Zhihua, "A Novel Bayes Model: Hidden Naive Bayes," *IEEE Transactions on Knowledge and Data Engineering*, vol. 21, pp. 1361-1371, 2009.
- [8] R. A. Baldwin, "Use of maximum entropy modeling in wildlife research," *Entropy*, vol. 11, pp. 854-866, 2009.
- [9] M. A. Hearst, S. T. Dumais, E. Osman, J. Platt, and B. Scholkopf, "Support vector machines," *Intelligent Systems and their Applications*, IEEE, vol. 13, pp. 18-28, 1998.
- [10] N. Cristianini and J. Shawe-Taylor, *An introduction to support vector machines and other kernel-based learning methods*: Cambridge university press, 2000.
- [11] S. R. Yerva, Z. Miklos, and K. Aberer, "It was easy, when apples and blackberries were only fruits," presented at the Third Web People Search Evaluation Forum (WePS-3), 2010.
- [12] C. Troussas, M. Virvou, K. Junshean Espinosa, K. Llaguno, and J. Caro, "Sentiment analysis of Facebook statuses using Naive Bayes classifier for language learning," in 4th International Conference on Information, Intelligence, Systems and Applications (IISA), 2013, pp. 1-6.

- [13] G. Salton, The SMART retrieval system : experiments in automatic document processing: Prentice-Hall, Inc., 1971.
- [14] F. Rosenblatt, The perceptron, a perceiving and recognizing automaton Project Para: Cornell Aeronautical Laboratory, 1957.
- [15] R. Rojas, "Perceptron Learning," in Neural Networks: A Systematic Introduction, ed: Springer Berlin Heidelberg, 1996, pp. 77-99.
- [16] C. E. Ortiz, "Introduction to Facebook APIs," ed: <http://www.ibm.com/>, 2010, pp. 1-20.
- [17] T. Wilson, Z. Kozareva, P. Nakov, A. Ritter, S. Rosenthal, and V. Stoyanov, "SemEval-2013 Task 2: Sentiment Analysis in Twitter," in Proceedings of the 7th International Workshop on Semantic Evaluation (SemEval), 2013.
- [18] S. Bird, E. Klein, and E. Loper, Natural language processing with Python: O'Reilly, 2009.
- [19] M. Hu and B. Liu, "Mining and summarizing customer reviews," presented at the Proceedings of the tenth ACM Special Interest Group on Knowledge Discovery and Data Mining (SIGKDD) international conference on Knowledge discovery and data mining, Seattle, WA, USA, 2004.
- [20] T. Wilson, P. Hoffmann, S. Somasundaran, J. Kessler, J. Wiebe, Y. Choi, et al., "OpinionFinder: a system for subjectivity analysis," presented at the Proceedings of HLT/EMNLP on Interactive Demonstrations, Vancouver, British Columbia, Canada, 2005.
- [21] F. Å. Nielsen, "A new ANEW: Evaluation of a word list for sentiment analysis in microblogs," presented at the 7th International Conference Mechatronic Systems and Materials (MSM 2011), Kaunas, Lithuania, 2011.
- [22] M. Elangovan, K. I. Ramachandran, and V. Sugumaran, "Studies on Bayes classifier for condition monitoring of single point carbide tipped tool based on statistical and histogram features," Expert Systems with Applications, vol. 37, pp. 2059-2065, 2010.
- [23] A. Cufoglu, M. Lohi, and K. Madani, "Classification accuracy performance of Naïve Bayesian (NB), Bayesian Networks (BN), Lazy Learning of Bayesian Rules (LBR) and Instance-Based Learner (IB1) - comparative study," in International Conference on Computer Engineering & Systems (ICCES), 2008, pp. 210-215.
- [24] P. Melville, W. Gryc, and R. D. Lawrence, "Sentiment analysis of blogs by combining lexical knowledge with text classification," presented at the Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining, Paris, France, 2009.
- [25] B. Yuan, Y. Liu, H. Li, T. T. T. PHAN, G. Kausar, C. N. Sing-Bik, et al., "Sentiment Classification in Chinese Microblogs: Lexicon-based and Learning-based Approaches," International Proceedings of Economics Development and Research (IPEDR) vol. 68, 2013.
- [26] E. Refaee and V. Rieser, "An Arabic Twitter Corpus for Subjectivity and Sentiment Analysis," in In 9 th International Conference on Language Resources and Evaluation (LREC'14), 2014.

- [27] L. Wang and C. Cardie, "A Piece of My Mind: A Sentiment Analysis Approach for Online Dispute Detection," in The 52nd Annual Meeting of the Association for Computational Linguistics Baltimore, USA, 2014.
- [28] D. H. Wolpert, "Stacked generalization," Neural networks, vol. 5, pp. 241-259, 1992.



## Using Arbiter and Combiner Tree to Classify Contexts of Data

Tawunrat Chalothorn and Jeremy Ellman

**Abstract—** This paper reports on the use of ensemble learning to classify as either positive or negative the sentiment of Tweets. Tweets were chosen as Twitter is a popular tool and a public, human annotated dataset was made available as part of the SemEval 2013 competition. We report on a classification approach that contrasts single machine learning algorithms with a combination of algorithms in an ensemble learning approach. The single machine learning algorithms used were Support Vector Machine (SVM) and Naïve Bayes (NB), while the methods of ensemble learning include the arbiter tree and the combiner tree. Our system achieved an F-score using Tweets and SMS with the arbiter tree at 83.57% and 93.55%, respectively, which was better than base classifiers; meanwhile, the results from the combiner tree achieved lower scores than base classifiers.

**Index Terms—** Tweets, contexts, positive, negative, natural language processing, ensemble learning, sentiment analysis

### I. Introduction

The research area of natural language processing (NLP) comprises various tasks; one of which is sentiment analysis. The main goal of sentiment analysis is to identify the polarity of natural language text. Sentiment analysis can be referred to as opinion mining; studying opinions, appraisals and emotions towards entities, events and their attributes. Sentiment analysis is a popular research area in NLP that aims to identify opinions or attitudes in terms of polarity. Currently, Twitter is a popular microblogging tool where users are increasing by the minute. Twitter allows users to post messages of up to 140 characters each time. These are called ‘Tweets’, which are often used to convey opinions about different topics. Consequently, various researchers are interested in classifying Tweets using sentiment analysis.

This paper introduces the original process of using the arbiter tree (Chan and Stolfo, 1993) and combiner tree (Chan and Stolfo, 1997), to classify the contexts of Tweet datasets and uses SMS datasets to evaluate the system. Arbiter tree (Chan and Stolfo, 1993) and combiner tree (Chan and Stolfo, 1997) have been chosen because they have not yet been used in sentiment analysis to classify Tweets or SMS datasets. The basic idea is to divide the training data into subsets, apply the learning algorithm to each and merge the resulting inducers. The main task is to find a solution to combining the appropriate learning model in order to achieve better results. Our main contribution is to propose and experiment with a combination of two machine learning algorithms, based on the use of the arbiter tree (Chan and Stolfo, 1993). The remainder of this paper is constructed as follows: the details of related works are mentioned in section 2. The corpus used is discussed in section 3; the methodology with data pre-processing and details of classifier are presented in section 4; section 5 discusses the details of the experiment and results. Finally, a conclusion and recommendations for future work are provided in section 6.

### II. Related Works

The microblogging tool Twitter is well-known and increasingly popular. The site allows users to post messages, or ‘Tweets’, of up to 140 characters each time. These are available for immediate download over the Internet. Tweets are extremely interesting to the marketing sector, since their rapid public interaction can indicate either customer success or presage public relations disasters far more quickly than web pages or traditional media.

Consequently, the content of Tweets and identifying their sentiment polarity as positive or negative is currently an active research topic. There are various researches that use Tweets with machine learning algorithms; for example, (Go *et al.*, 2009) classify Twitter using Naïve Bayes (NB) (Lewis, 1998; Liangxiao *et al.*, 2009), Maximum Entropy Modelling (Jaynes, 1957; Baldwin, 2009) and Support Vector Machine (SVM) (Hearst *et al.*, 1998; Cristianini and Shawe-Taylor, 2000). In the experiment, emoticons have been used as noisy labels in training data to identify the label as positive or negative. Emoticons can be referred to printable characters of emotion, such as :- ) for smile and :-( for sad. SVM (Hearst *et al.*, 1998; Cristianini and Shawe-Taylor, 2000) with unigram obtained high accuracy at 82.90%. (Go *et al.*, 2009) note that using negation and part-of-speech tagging did not help improve accuracy.

(Aisopos *et al.*, 2011) divided Tweets into three groups using emoticons for classification. If Tweets contain positive emoticons, they will be classified as positive, and vice versa. Other Tweets that do not have positive/negative emoticons will be classified as neutral. However, those that contain both positive and negative emoticons are ignored in their study. Their task focused on analyzing the contents of social media using n-gram graphs. The results revealed that n-grams yielded high accuracy when tested with C4.5 (Abdel-Dayem, 2010), but low accuracy with NB Multinomial (NBM) (McCallum and Nigam, 1998b).

### III. Corpus

The datasets used in our experiment are taken from SemEval 2013 (Wilson *et al.*, 2013). The data were gathered from Twitter; a well-known and increasingly popular microblogging site. Twitter allows its users to post messages, or 'Tweets', of up to 140 characters each time, which are available for immediate download over the Internet. Tweets are extremely interesting in marketing terms, since their rapid public interaction can either indicate customer success or presage public relations

disasters far more quickly than web pages or traditional media. Consequently, the content of tweets and identifying their sentiment polarity as positive or negative is a current active research topic.

The datasets comprise training data, testing data and gold standard. Gold standard refers to the testing data labelled with the correct polarity. However, these datasets were annotated using five Mechanical Turk workers; also known as Turkers (Wilson *et al.*, 2013). For each sentence, they will use the start and end point of their opinion for the phrase or word, and state whether it is negative, neutral or positive. Then, the words that appear three times from five votes will be assigned the label. In addition to Tweets, SMS messages are used to evaluate the system. SMS messages are also obtained from the organizer of SemEval 2013 (Wilson *et al.*, 2013). Only the datasets labelled as positive and negative will be used in this research.

Furthermore, three sentiment lexicons were used in this experiment. They are Bing Liu Lexicon (HL) (6780 words), collected over many years by (Hu and Liu, 2004). They began to accumulate lexicons in 2004, during the course of their work on online customer product reviews (Hu and Liu, 2004). MPQA Subjective Lexicon (MPQA) (8221 words) was created by (Wilson *et al.*, 2005a) using a set of approximately 400 documents. AFINN Lexicon (AFINN) (2477 words) was created from Twitter between 2009-2011 by (Nielsen, 2011a) for use in the United Nation Climate Conference (COP15).

### IV. Methodologies

#### A. Data pre-processing

For the process of data pre-processing, emoticons were labelled by matching those collected manually from the dataset against a well-known group of emoticons. Subsequently, negative contractions were expanded and converted to full form (e.g. don't -> do not). Moreover, the features of Tweets were removed or replaced by words, such as Twitter usernames, URLs and hashtags.

A Twitter username is a unique name displayed in the user's profile and may be used for both authentication and identification. This is demonstrated by prefacing the username with an @ symbol. When a Tweet is directed towards a specific individual or entity, this can be displayed by including @username in the Tweet. For example, a Tweet directed at 'som' would include the text @som. Before URLs are posted to Twitter, they are shortened automatically to use the t.co domain whose modified URLs contain a maximum of 22 characters. However, both features have been removed from the datasets. Hashtags are used to represent keywords and topics in Twitter by using # followed by words or phrases; for example, #newcastleuk. This feature has been replaced with the following word after the # symbol. For example, #newcastleuk was replaced with newcastleuk.

Frequently, repeated letters are used to provide emphasis in Tweets. These were reduced and replaced using a simple regular expression by two of the same characters. For example, happpppppy will be replaced with happy, and coollllll will be replaced with cool. Next, special characters were removed, such as [, {, ?, and !. Slang and contracted words were converted to their full form; for example, 'fyi' became 'for your information'. Finally, Natural Language Toolkit (NLTK) (Bird *et al.*, 2009b) stopwords were removed from the datasets, such as 'a', 'the', etc.. The metric and comparison of these features can be found in (Chalothorn and Ellman, 2014). The flowchart of data processing are shown in Fig. 1.

### B. Arbiter Trees

Arbiter tree (Chan and Stolfo, 1993) is a method that uses training data classified by using base classifiers with selection rules. Selection rules are used to compare the prediction of base classifiers for choosing the training dataset for the arbiter. Then, the final prediction is decided based on the base classifiers and arbiter by using

arbitration rules with the aim of learning from incorrect classification (Chan and Stolfo, 1993).

### C. Combiner Tree

The Combiner tree (Chan and Stolfo, 1997) method has similar qualities to the arbiter tree but it will be trained directly by the training output from the base classifiers that passed the composition rules. Next, the final prediction will be classified by the combiner. There are two versions of composition rules: the first uses the combination of results from the base classifier; while the second uses the same as the first with the addition of training data attributes. The aim of the combiner tree is to learn from correct classification (Chan and Stolfo, 1997)

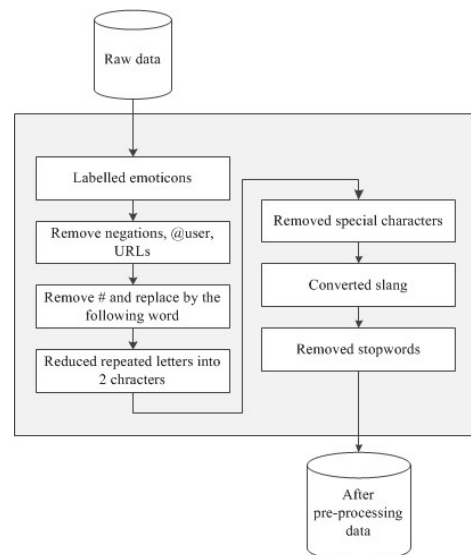


Fig. 6. Fig. 1. Flowchart of data pre-processing

### D. Support Vector Machine

SVM (Hearst *et al.*, 1998; Cristianini and Shawe-Taylor, 2000) is a binary linear classification model with the learning algorithm for classification and regression analysis of data, and recognizing the pattern. The purpose of SVM is to separate datasets into classes and discover the decision boundary (hyper-plane). To find the hyper-plane, the maximum distance between classes (margin) will be used with the closest data points on the margin (support vector). The equation of SVM can present as:

$$\vec{w} = \sum_j \alpha_j c_j \vec{d}_j, \quad \alpha_j \geq 0 \quad (1)$$

where vector  $\vec{w}$  represented as hyperplane.  $c_j$  is a polarity (negative and positive) of the data  $d_j$  which  $c_j \in \{-1, 1\}$ .  $\alpha_j$  are obtained by solving the dual optimisation problem. Those  $\vec{d}_j$  such that  $\alpha_j$  is greater than zero are called, support vectors, since they are the only document vectors contributing to  $\vec{w}$ . Classification of test instances consists simply of determining which side of  $\vec{w}$  hyperplane they fall on. Our research used the default setting of SVMLight for the SVM classifier model. SVMLight is an implementation of SVM in C.

#### E. Naïve Bayes

The NB algorithm (Liangxiao *et al.*, 2009) is a classification algorithm based on Bayes' theorem that underlies the naïve assumption that attributes within the same case are independent given the class label (Elangovan *et al.*, 2010). This is also known as the state-of-art Bayes rules (Cufoglu *et al.*, 2008). NB (Liangxiao *et al.*, 2009) constructs the model by adjusting the distribution of the number for each feature. For example, in text classification, NB regards the documents as a bag-of-words, from which it extracts features. NB (Liangxiao *et al.*, 2009) model follows the assumption that attributes within the same case are independent given the class label (Hope and Korb, 2004). Tang *et al.* (2009) considered that Naïve Bayes assigns a context  $X_i$  (represented by a vector  $X_i^*$ ) to the class  $C_j$  that maximizes  $P(C_j|X_i^*)$  by applying Bayes's rule, as in (2).

$$P(C_j|X_i^*) = \frac{P(C_j)P(X_i^*|C_j)}{P(X_i^*)} \quad (2)$$

where  $P(X_i^*)$  is a randomly selected context  $X$ . The representation of vector is  $X_i^*$ .  $P(C)$  is the random select context that is assigned to class  $C$ .

To classify the term  $P(X_i^*|C_j)$ , features in  $X_i^*$  were assumed as  $f_j$  from  $j = 1$  to  $m$  as in (3).

$$P(C_j|X_i^*) = \frac{P(C_j) \prod_{j=1}^m P(f_j|C_j)}{P(X_i^*)} \quad (3)$$

In this research, the NB algorithm was used from the NLTK, which is a widely-used machine learning algorithm, open source, developed using Python and comprising the WordNet interface.

#### V. Experiment and results

In our experiment, the idea from (Chan and Stolfo, 1993) has been adapted using the arbiter tree algorithm, as only two classifiers are used with one training data. In order to build the training data, all selection rules from (Chan and Stolfo, 1993) were adapted and used in this experiment. The processes for creating training data are detailed below:

- 1) Base training data were trained into base classifiers, which are SVM (Hearst *et al.*, 1998; Cristianini and Shawe-Taylor, 2000) and NB (Liangxiao *et al.*, 2009). The base training data were yielded from the combination of the sentiment lexicons noted in section III. They were combined by removing the words that duplicate, overlap and contradict in sentiment (Melville *et al.*, 2009; Yuan *et al.*, 2013; Refaee and Rieser, 2014; Wang and Cardie, 2014).
- 2) After obtaining the results from the base classifiers, they were united and passed into selection rules. There are three versions of selection rules:
  - i. Selection rule 1 is the different results from classifiers 1 and 2.
  - ii. Selection rule 2 is the union of the results from selection rule 1 and the results from classifiers 1 and 2, which are the same prediction but incorrect.
  - iii. Selection rule 3 is the union of selection rules 1 and 2 and the

results of classifiers 1 and 2, which are the same prediction and correct.

- 3) As in the arbiter tree algorithm, (Chan and Stolfo, 1993) did not state clearly how to use the selection rules; therefore, the data from selection rules 1, 2 and 3 have been trained with base classifiers that assume to be the arbiter for creating the final training data. The flowchart of these processes is presented in Fig. 2.

After obtaining the final training data for the arbiter, they were used in the final classification process for the final prediction results. During this process (see Fig. 3), the base classifiers were trained by using base training data, while the arbiter was trained by using arbiter training data to classify the test set. Next, their results went through the process of arbiter rules for the final prediction results. There are two versions of arbiter rules. The first uses the majority vote of prediction from the base classifier and the arbiter prediction. If the results of predictions 1 and 2 are equal, the results from prediction 2 will be used. Conversely, the arbiter results will be used. In the second version, if the results of predictions 1 and 2 are not equal, the different arbiter results will be used. If the results of prediction 1 are equal to those of the correct arbiter, use the correct arbiter results. In contrast, the incorrect results from the arbiter tree were used. The evaluation metric was used F-score (Powers, 2011).

The datasets of Tweets and SMS were tested in the arbiter tree (Chan and Stolfo, 1993). Their results are presented in Table 1. Following the comparison between the arbiter and base classifiers (Table 2), the results of Tweets using arbiter rules version 1 did not achieved better accuracy than base classifiers at 82.31%; meanwhile, the results from arbiter rules version 2 achieved a better F-score than SVM (Hearst *et al.*, 1998; Cristianini and Shawe-Taylor, 2000) and NB (Liangxiao

*et al.*, 2009) at 83.57 %. Conversely, the results of the SMS dataset revealed that the results from arbiter rule version 1 and 2 achieved a better F-score than base classifiers at 84.57% and 85.56%, respectively.

In addition to the arbiter tree (Chan and Stolfo, 1993), the combiner tree (Chan and Stolfo, 1997) was also used in the experiment for comparison purposes. The training dataset for the combiner have to be built based on the base classifiers and composition rules, see Fig. 4. There are two versions of the composition rules: The first version uses the combination of results from the base classifiers, while the second uses a combination of the first version and the instance from training data. Next, they will be used as the training data for classify the testing data. The results of testing Tweets demonstrated a very low F-score of 30.25% and 32.36% respectively for the first and second versions. Conversely, the results from SMS revealed F-scores of 34.59% and 34.65% respectively for the first and second versions. The results from the combiner tree [2] (see Table III) achieved lower F-scores than base classifiers in both datasets.

## VI. Conclusion and Future Work

In this experiment, the original process of using the arbiter tree (Chan and Stolfo, 1993) and combiner tree (Chan and Stolfo, 1997) algorithms to classify Tweets and SMS datasets have been demonstrated and clearly explained. The use of ensemble learning might not always have achieved the most accuracy as the results from combiner tree (Chan and Stolfo, 1997); however, the results of the classification of Tweets and SMS dataset using arbiter tree (Chan and Stolfo, 1993), demonstrated their ability to achieve F-scores of 83.57% and 85.56%, respectively, which is better than the scores achieved for both base classifiers.

For future work, the results from the arbiter tree (Chan and Stolfo, 1993) will be combined with the SVM (Hearst *et al.*, 1998; Cristianini and Shawe-Taylor,

2000), NB (Liangxiao *et al.*, 2009) and SentiStrength (Thelwall *et al.*, 2010b) by using majority voting. The main purpose is to improve sentiment classification using a combination of machine learning algorithms and sentiment resources. SentiStrength (Thelwall *et al.*, 2010b) is the sentiment analysis methodology used to judge whether a sentence has a positive or negative sentiment, which is developed from comments posted on MySpace.

TABLE I: THE RESULTS OF TWEETS AND SMS DATASET FROM BASE CLASSIFIERS

	<b>Tweet dataset</b> Avg. F-score (%)	<b>SMS dataset</b> Avg. F-score (%)
SVM	83.55	85.49
NB	81.54	85.05

TABLE II: THE RESULTS OF TWEETS AND SMS DATASET FROM ARBITER TREE

	<b>Tweet dataset</b> Avg. F-score (%)	<b>SMS dataset</b> Avg. F-score (%)
Arbiter rules version 1	82.31	84.87
Arbiter rules version 2	83.57	85.56

TABLE III: THE RESULTS OF TWEETS AND SMS DATASET FROM COMBINER TREE

	<b>Tweet dataset</b> Avg. F-score (%)	<b>SMS dataset</b> Avg. F-score (%)
Combiner rules version 1	30.25	34.59
Combiner rules version 2	32.36	34.65

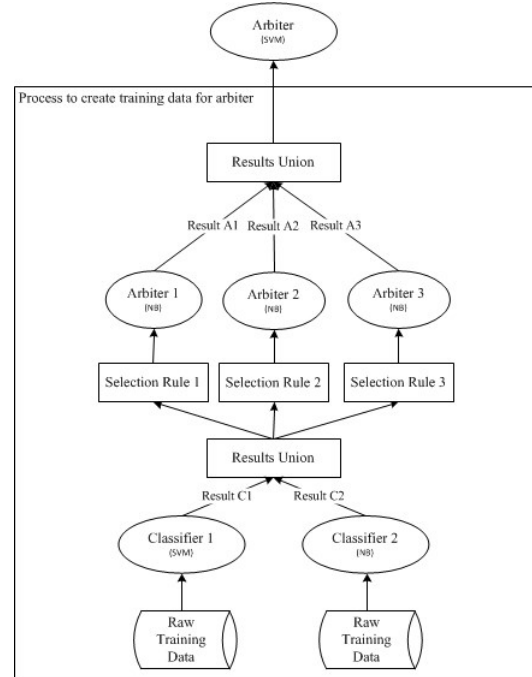


Fig. 7. Fig. 2. Process for making training data for arbiter

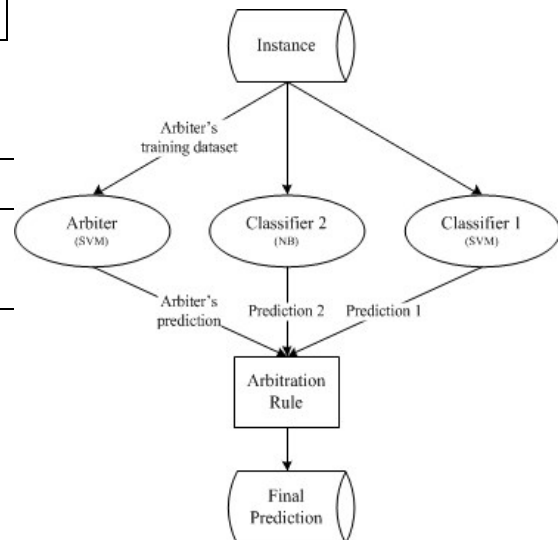


Fig. 8. Fig. 3. Process for final prediction of the testing data of arbiter tree

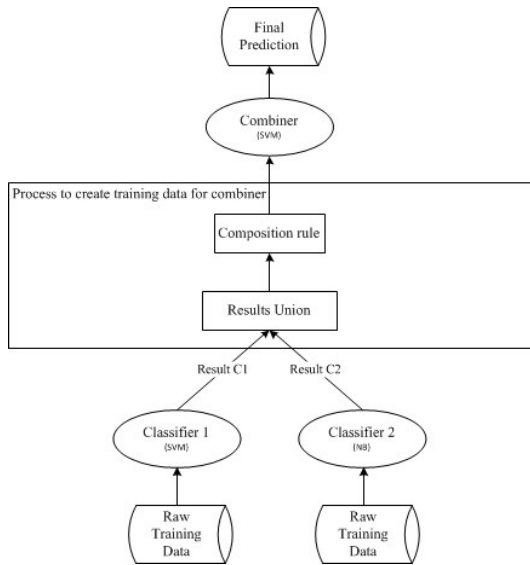


Fig. 9. Fig. 4. Process of combiner tree

#### REFERENCES

- [1] P. K. Chan and S. J. Stolfo, "Toward parallel and distributed learning by meta-learning," presented at the The International Association for the Advancement of Artificial Intelligence (AAAI) workshop in Knowledge Discovery in Databases, 1993.
- [2] P. K. Chan and S. J. Stolfo, "On the accuracy of meta-learning for scalable data mining," *Journal of Intelligent Information Systems*, vol. 8, pp. 5-28, 1997.
- [3] A. Go, R. Bhayani, and L. Huang, "Twitter sentiment classification using distant supervision," *CS224N Natural Language Processing, Project Report*, Stanford, pp. 1-12, 2009.
- [4] D. D. Lewis, "Naive (Bayes) at Forty: The Independence Assumption in Information Retrieval," presented at the Proceedings of the 10th European Conference on Machine Learning, 1998.
- [5] J. Liangxiao, H. Zhang, and C. Zhihua, "A Novel Bayes Model: Hidden Naive Bayes," *IEEE Transactions on Knowledge and Data Engineering*, vol. 21, pp. 1361-1371, 2009.
- [6] E. T. Jaynes, "Information theory and statistical mechanics," *Physical review*, vol. 106, p. 620, 1957.
- [7] R. A. Baldwin, "Use of maximum entropy modeling in wildlife research," *Entropy*, vol. 11, pp. 854-866, 2009.
- [8] M. A. Hearst, S. T. Dumais, E. Osman, J. Platt, and B. Scholkopf, "Support vector machines," *IEEE Intelligent Systems and their Applications*, vol. 13, pp. 18-28, 1998.
- [9] N. Cristianini and J. Shawe-Taylor, *An introduction to support vector machines and other kernel-based learning methods*: Cambridge university press, 2000.
- [10] F. Aisopos, G. Papadakis, and T. Varvarigou, "Sentiment analysis of social media content using N-Gram graphs," presented at the Proceedings of the 3rd ACM the Special Interest Group on Multimedia (SIGMM) international workshop on Social media, Scottsdale, Arizona, USA, 2011.
- [11] A. R. Abdel-Dayem, "Detection of arterial lumen in sonographic images based on active contours and diffusion filters," presented at the Proceedings of the 7th international conference on Image Analysis and Recognition - Volume Part II, Portugal, 2010.
- [12] A. McCallum and K. Nigam, "A comparison of event models for Naive Bayes text classification," in *IN AAAI-98 WORKSHOP ON LEARNING FOR TEXT CATEGORIZATION*, 1998, pp. 41-48.
- [13] T. Wilson, Z. Kozareva, P. Nakov, A. Ritter, S. Rosenthal, and V. Stoyanov, "SemEval-2013 Task 2: Sentiment Analysis in Twitter," in *Proceedings of the 7th International Workshop on Semantic Evaluation (SemEval)*, 2013.
- [14] M. Hu and B. Liu, "Mining and summarizing customer reviews," presented at the Proceedings of the tenth ACM

Special Interest Group on Knowledge Discovery and Data Mining (SIGKDD) international conference on Knowledge discovery and data mining, Seattle, WA, USA, 2004.

- [15] T. Wilson, P. Hoffmann, S. Somasundaran, J. Kessler, J. Wiebe, Y. Choi, et al., "OpinionFinder: a system for subjectivity analysis," presented at the Proceedings of HLT/EMNLP on Interactive Demonstrations, Vancouver, British Columbia, Canada, 2005.
- [16] F. A. Nielsen, "A new ANEW: Evaluation of a word list for sentiment analysis in microblogs," presented at the 7th International Conference Mechatronic Systems and Materials (MSM 2011), Kaunas, Lithuania, 2011.
- [17] S. Bird, E. Klein, and E. Loper, Natural language processing with Python: O'Reilly, 2009.
- [18] T. Chalothorn and J. Ellman, "TJP: Identifying the Polarity of Tweets from Contexts," presented at the Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014), Dublin, Ireland, 2014.
- [19] M. Elangovan, K. I. Ramachandran, and V. Sugumaran, "Studies on Bayes classifier for condition monitoring of single point carbide tipped tool based on statistical and histogram features," Expert Systems with Applications, vol. 37, pp. 2059-2065, 2010.
- [20] A. Cufoglu, M. Lohi, and K. Madani, "Classification accuracy performance of Naive Bayesian (NB), Bayesian Networks (BN), Lazy Learning of Bayesian Rules (LBR) and Instance-Based Learner (IB1) - comparative study," presented at the International Conference on Computer Engineering & Systems (ICCES), 2008.
- [21] L. R. Hope and K. B. Korb, "A bayesian metric for evaluating machine

learning algorithms," presented at the Proceedings of the 17th Australian joint conference on Advances in Artificial Intelligence, Cairns, Australia, 2004.

- [22] H. Tang, S. Tan, and X. Cheng, "A survey on sentiment detection of reviews," Expert Systems with Applications, vol. 36, pp. 10760-10773, 2009.
- [23] P. Melville, W. Gryc, and R. D. Lawrence, "Sentiment analysis of blogs by combining lexical knowledge with text classification," presented at the Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining, Paris, France, 2009.
- [24] B. Yuan, Y. Liu, H. Li, T. T. T. PHAN, G. Kausar, C. N. Sing-Bik, et al., "Sentiment Classification in Chinese Microblogs: Lexicon-based and Learning-based Approaches," International Proceedings of Economics Development and Research (IPEDR) vol. 68, 2013.
- [25] E. Refaee and V. Rieser, "An Arabic Twitter Corpus for Subjectivity and Sentiment Analysis," in In 9 th International Conference on Language Resources and Evaluation (LREC'14), 2014.
- [26] L. Wang and C. Cardie, "A Piece of My Mind: A Sentiment Analysis Approach for Online Dispute Detection," in The 52nd Annual Meeting of the Association for Computational Linguistics Baltimore, USA, 2014.
- [27] D. M. Powers, "Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation," Machine Learning RTechnologies, vol. 2, pp. 37-63, 2011.
- [28] M. Thelwall, K. Buckley, G. Paltoglou, D. Cai, and A. Kappas, "SentiStrength," ed: University of Wolverhampton, 2010.





Publication 9: The International Conference on Information Science & Applications (ICISA 2015)

## Simple approaches of sentiment analysis via ensemble learning

Tawunrat Chalothorn and Jeremy Ellman

Department of Computer Science and Digital Technologies  
University of Northumbria at Newcastle, Pandon Building,  
Camden Street Newcastle Upon Tyne, NE2 1XE, United Kingdom  
{tawunrat.chalothorn,jeremy.ellman}@northumbria.ac.uk

**Abstract.** Twitter has become a popular microblogging tool where users are increasing every minute. It allows its users to post messages of up to 140 characters each time; known as 'Tweets'. Tweets have become extremely attractive to the marketing sector, since the user can either indicate customer success or presage public relations disasters far more quickly than web pages or traditional media. Moreover, the content of Tweets has become a current active research topic on sentiment polarity as positive or negative. Our experiment of sentiment analysis of contexts of tweets show that the accuracy performance can improve and be better achieved using ensemble learning, which is formed by the majority voting of the Support Vector Machine, Naive Bayes, SentiStrength and Stacking.

**Keywords:** Twitter, Tweet, sentiment, analysis, natural language processing, ensemble learning.

### 1 Introduction

Natural language processing (NLP) is a research area composed of various tasks; one of which is sentiment analysis. The main goal of sentiment analysis is to identify the polarity of natural language text (Shaikh *et al.*, 2007). Sentiment analysis can be referred to as opinion mining; studying opinions, appraisals and emotions towards entities, events and their attributes (Pang and Lee, 2008). Sentiment analysis is a popular research area in NLP, which aims to identify opinions or attitudes in terms of polarity. Consequently, various researchers are interested in classifying Tweets using sentiment analysis. Many studies focus on using a single classifier, such as Naive Bayes and Support Vector Machine (SVM), to analyze sentiment. However, this paper demonstrates that the use of multiple classifiers in ensemble learning can improve the performance accuracy of sentiment classification. Moreover, we investigate the use of sentiment lexicons that could affect the classification.

The main contribution can be broken down as follows: (i) the ensemble classifiers have been formed using supervised and semi-supervised learning; (ii) sentiment lexicons and bag-of-words (BOW) have been combined for the comparison and clearly shown; (iii) the combinations of lexicons and BOW for use in supervised, semi-supervised and ensemble learning are explained and discussed. The remainder of this paper is constructed as follows: related work is discussed in section 2; the methodology, experiment and results are presented in sections 3 and 4, respectively. Finally, a conclusion and recommendations for future work are provided in section 5.

### 2 Related works

Twitter is a popular social networking and microblogging site that allows users to post messages of up to 140 characters; known as 'Tweets'. Tweets are extremely attractive to the marketing sector, since they can be searched in real-time. The word 'emoticon' is a neologistic contraction of 'emotional icon'. Specifically, it refers to the combination of punctuation characters to indicate sentiment in a text. Well-known emoticons include :) to represent a happy face, and :( a sad one.

Emoticons allow writers to augment the impact of limited texts (such as in SMS messages or tweets) using fewer characters. (Go *et al.*, 2009) used supervision to classify sentiment of Twitter. Emoticons were used as noisy labels in training data; thereby facilitating the performance of supervised learning (positive and negative) at a distance. Three classifiers were used: Naïve Bayes, Maximum Entropy and SVM. Respectively, these classifiers were able to obtain more than 81.30%, 80.50% and 82.20% accuracy on their unigram testing data.

Moreover, (Gryc and Moilanen, 2014) used stacking and majority voting to analyse sentiment of the dataset obtained from IBM's Predictive Modelling Group. The datasets are concerned with posts related to the 2008 U.S. presidential election. The datasets were labelled as positive, neutral and negative by the service of Amazon Mechanical Turk. However, only positive, neutral and negative labels were used. Three features were used in the experiment: social network features, sentiment analysis features and unigram BOW features. The use of each feature was separated into four sections: social network features used with Logistic Regression, named SNA; sentiment analysis feature used with NBM, named SA; NBM used with unigram BOW features, named as BOW; and NBM used with all features and named as ALL. Next, two ensemble learning called majority voting and stacking were used with the first three sections. The results showed that they achieved F-scores of 36.30%, 44.63%, 48.41% and 47.71% for SNA, SA, BOW and ALL, respectively. Conversely, stacking and majority voting achieved F-scores of 44.33% and 46.68%, respectively. In the comparison, stacking and majority voting achieved lower F-scores than BOW and ALL.

### 3 Methodologies

#### 3.1 Classifier

Two machine learning, one sentiment resource and two ensemble learning are used in this research and are detailed below.

**Support Vector Machine (SVM)** (Cristianini and Shawe-Taylor, 2000) was used from SVMLight. SVM is a binary linear classification model with the learning algorithm for classifying and regression analysing the data and recognising the pattern. The purpose of SVM is to separate datasets into classes and discover the decision boundary (hyper-plane). To find the hyper-plane, the maximum distance between classes (margin) will be used with the closest data points on the margin (support vector).

**Naïve Bayes (NB) algorithm** (Tan *et al.*, 2009) was used from NLTK. NB is a classification algorithm based on Bayes' theorem that underlies the naïve assumption that attributes within the same case are independent given the class label (Elangovan *et al.*, 2010). This is also known as the state-of-art of Bayes rules (Cufoglu *et al.*, 2008). NB (Tan *et al.*, 2009) constructs the model by adjusting the distribution of the number for each feature. For example, in the text classification, NB (Tan *et al.*, 2009) regards the documents as a BOW and from which it extracts features (Liu, 2007; 2012b). NB (Tan *et al.*, 2009) model follows the assumption that attributes within the same case are independent given the class label (Hope and Korb, 2004). Tang *et al.* (2009) considered that Naïve Bayes assigns a context  $X_i$  (represented by a vector  $X_i^*$ ) to the class  $C_j$  that maximizes  $P(C_j|X_i^*)$  by applying Bayes's rule, as in (1).

$$P(C_j|X_i^*) = \frac{P(C_j)P(X_i^*|C_j)}{P(X_i^*)} \quad (1)$$

where  $P(X_i^*)$  is a randomly selected context  $X$ . The representation of vector is  $X_i^*$ .  $P(C)$  is the random select context that is assigned to class  $C$ .

To classify the term  $P(X_i^*|C_j)$ , features in  $X_i^*$  were assumed as  $f_j$  from  $j = 1$  to  $m$  as in (2).

$$P(C_j|X_i^*) = \frac{P(C_j) \prod_{j=1}^m P(f_j|C_j)}{P(X_i^*)} \quad (2)$$

**Majority voting** (Polikar, 2012) or called, majority rules are basic and simple algorithm that uses the combination of various classifiers. The decisions of the voting are depended on agreement among more than half of the classifiers otherwise the input is rejected. The equation of majority voting (Polikar, 2012) can presented as:

$$\sum_{i=1}^L d_{i,k} = \max_{j=1,\dots,c} \sum_{i=1}^L d_{i,j} \quad (3)$$

where it is assumed that the label outputs of the classifiers are given as  $c$  dimensional binary vectors (for majority rules only two classes, i.e.  $[d_{i,1}, d_{i,2}]^T \in \{0,1\}^c, i = 1, \dots, L$ ), and where  $d_{i,j} = 1$  if  $D_i$  labels  $x$  in  $w_j$  and 0 otherwise.

**Stacking (ST)** (Wolpert, 1992) was used from WEKA. ST is technique that uses the prediction of the base learning algorithms as a training data to produce the final prediction, whereby the ST techniques can be represented by any learning algorithm. Meanwhile, **SentiStrength (SS)** (Thelwall *et al.*, 2010b) is also available to use free of charge and has been adopted by some researchers. SentiStrength is the analysis methodology used to judge whether a sentence has a positive or negative sentiment. The methodology was developed by (Thelwall *et al.*, 2010a), using nearly 4,000 comments on MySpace. They used three annotators and Krippendorff's alpha (Krippendorff, 1980) to measure their agreement. The data have been separated into two groups: trail data and testing data. Trail data was used to identify algorithms for judgement and suitable scales. Algorithms were identified, ranging from 1 to 5, and used alongside testing data for final judgement. These will be SentiStrength's lexicon.

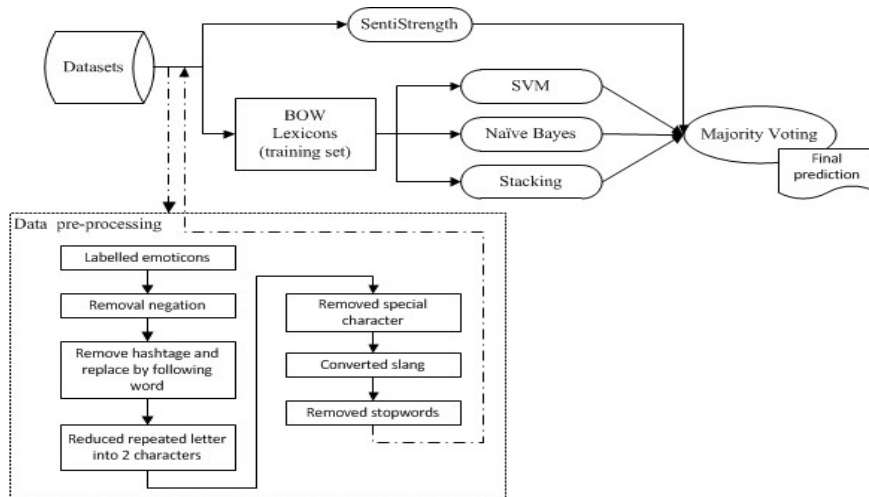


Figure 1. Flowchart of our approach in the experiment

### 3.2 Pre-processing

The datasets used in our experiment are from SemEval 2013 Task 2A (Wilson *et al.*, 2013). For data pre-processing, emoticons were labelled by matching those collected manually from the dataset against a well-known collection of emoticons. Subsequently, negative contractions were expanded in place and converted to full form (e.g. don't -> do not). Moreover, the features of Twitter were also removed or replaced by words, such as twitter usernames, URLs and hashtags. A Twitter username is a unique name displayed in the user's profile and may be used for both authentication and identification. This is shown by prefacing the username with an @ symbol. When a Tweet is directed at an individual or particular entity, this can be shown in the tweet by including @username. For example, a Tweet directed at 'som' would include the text @som. Before URLs are posted to Twitter, they are shortened automatically to use the t.co domain, whose modified URLs are a maximum of 22 characters. However, both features have been removed from the datasets. Hashtags are used to represent keywords and topics in Twitter by using # followed by

words or phrases, such as #newcastleuk. This feature has been replaced with the following word after the # symbol. For example, #newcastleuk was replaced by newcastleuk. Frequently repeated letters are used to convey emphasis in Tweets. These were reduced and replaced using a simple regular expression by two of the same character. For example, happpppppy will be replaced with happy, and coollllll will be replaced by cool. Next, special characters were removed, such as [.,{,?,and !. Slang and contracted words were converted to their full form; for example, 'fyi' became 'for your information'. Finally, NLTK (Bird *et al.*, 2009a) stopwords were removed from the dataset, such as 'a', 'the', etc..

Moreover, three sentiment lexicons were used in this experiment. Bing Liu Lexicon (HL) (Hu and Liu, 2004), MPQA Subjective Lexicon (MPQA) (Wilson *et al.*, 2005a) and AFINN Lexicon (AFINN) (Nielsen, 2011b).

#### 4 Experiments and results

The experiments were tested using SentiStrength, NB model in NLTK and SVMLight for individual classification. Both datasets used the same method of pre-processing. For ensemble learning, ST was used based on WEKA, and majority voting was implemented using Python. SentiStrength has been used as a server and accessed by my application in Python for passing the testing data directly to the SentiStrength website to calculate the testing data scores. The contents of BOW are from the training datasets of Tweets. The use of BOW has been tested against the combination of BOW and sentiment lexicons. These were merged by removing words that duplicate, overlap and contradict in sentiment.

They were tested using SVM and NB on two datasets: Tweets and SMS. The results showed that the F-score accuracy improved after being combined with BOW; reaching 83.55% for Tweets and 87.85% for SMS dataset, as illustrated in Table 1. For the processing of ensemble learning, known as stacking, the combination of SVM and NB as level 0 classifier and bagging as level 1 classifier. The training data used in ST was chosen from the combination of BOW and sentiment lexicons that obtained that highest F-score. For Tweet datasets, the training dataset was used from the combination of BOW, HL, MPQA and AFINN, which obtained the highest F-score of 83.55% from using the SVM classifier. Conversely, for SMS datasets, the training dataset was used from the combination of BOW and MPQA, which obtained the highest F-score of 87.85% from using the NB classifier. After testing both datasets in ST, he results demonstrated their ability to obtain F-scores of 84.05% and 85.57% for Twitter and SMS datasets, respectively. Next, majority voting was used for the combination of all classifiers. The combination used in majority voting was separated into two, three and four voters. There are problems in the first and third, as half of the voters are not equal. This problem could be solved by using two conditions from (Martin-Valdivia *et al.*, 2013). The first condition (V01), positive will be used to represent the answer if they are not equal, while negative has been used in the second condition (V02). The overall results (Table 1) demonstrate that the combination of three classifiers using majority voting achieved the highest score for both Tweets and SMS datasets. For Tweets, the combination of SVM, SentiStrength and ST achieved the highest F-score at 86.05%. Meanwhile, the combination of NB, SentiStrength and ST achieved the highest F-score at 88.82% for SMS dataset. Our system is quite good in comparison to the results of Tweets and SMS data; whereby both achieve F-scores of more than 85%.

**Table 1.** All results of Tweets and SMS dataset

Methods of Tweet dataset	Avg. F-score (%)	Methods of SMS dataset	Avg. F-score (%)
NB - BOW	81.94	NB - Bow	85.49
SVM - BOW	83.55	SVM - BOW	85.05
SS	78.37	SS	79.83
NB - BOW + HL	80.84	NB - BOW + HL	84.51
NB - BOW + HL + MPQA	81.26	NB - BOW + HL + MPQA	84.56
NB - BOW + HL + MPQA + AFINN	81.94	NB - BOW + HL + MPQA + AFINN	85.03
NB - BOW + HL + AFINN	81.74	NB - BOW + HL + AFINN	84.98
NB - BOW + MPQA	82.57	NB - BOW + MPQA	<b>87.85</b>
NB - BOW + MPQA + AFINN	81.73	NB - BOW + MPQA + AFINN	84.84
NB - BOW + AFINN	82.91	NB - BOW + AFINN	87.25
SVM - BOW + HL	82.47	SVM - BOW + HL	85.54
SVM - BOW + HL + MPQA	82.81	SVM - BOW + HL + MPQA	85.45
SVM - BOW + HL + MPQA + AFINN	<b>83.55</b>	SVM - BOW + HL + MPQA + AFINN	85.78
SVM - BOW + HL + AFINN	83.32	SVM - BOW + HL + AFINN	85.96
SVM - BOW + MPQA	81.99	SVM - BOW + MPQA	85.63
SVM - BOW + MPQA + AFINN	83.20	SVM - BOW + MPQA + AFINN	86.05
SVM - BOW + AFINN	83.00	SVM - BOW + AFINN	84.95
ST	84.05	ST	85.57
ENS (SVM + NB) (V01)	83.82	ENS (SVM + NB) (V01)	86.68
ENS (SVM + NB) (V02)	81.65	ENS (SVM + NB) (V02)	86.74
ENS (SVM + SS) (V01)	84.44	ENS (SVM + SS) (V01)	84.87
ENS (SVM + SS) (V02)	77.33	ENS (SVM + SS) (V02)	80.30
ENS (SVM + ST) (V01)	84.02	ENS (SVM + ST) (V01)	85.51
ENS (SVM + ST) (V02)	83.57	ENS (SVM + ST) (V02)	85.68
ENS (NB + SS) (V01)	83.30	ENS (NB + SS) (V01)	86.40
ENS (NB + SS) (V02)	76.82	ENS (NB + SS) (V02)	81.14
ENS (NB + ST) (V01)	83.46	ENS (NB + ST) (V01)	86.63
ENS (NB + ST) (V02)	82.39	ENS (NB + ST) (V02)	86.72
ENS (SS + ST) (V01)	82.33	ENS (SS + ST) (V01)	85.74
ENS (SS + ST) (V02)	79.66	ENS (SS + ST) (V02)	79.28
ENS (SVM + NB + SS)	84.09	ENS (SVM + NB + SS)	87.90
ENS (SVM + NB + ST)	84.28	ENS (SVM + NB + ST)	86.19
ENS (SVM + SS + ST)	<b>86.05</b>	ENS (SVM + SS + ST)	86.54
ENS (NB + SS + ST)	85.91	ENS (NB + SS + ST)	<b>88.82</b>
ENS (SVM + NB + SS + ST) (V01)	84.54	ENS (SVM + NB + SS + ST) (V01)	87.13
ENS (SVM + NB + SS + ST) (V02)	83.87	ENS (SVM + NB + SS + ST) (V02)	87.58

## 5 Conclusion and future work

In this research, the demonstration of using machine and ensemble learning formed by different components can provide state-of-the-art results for this particular domain. Moreover, we compared the use of BOW with the combination of lexicon and BOW. The results showed that the F-score of the combination of BOW and sentiment lexicons achieved greater accuracy than using only BOW. Furthermore, the results demonstrate that the size of training data does not always affect performance accuracy, provided they did not have sufficient information related to the test data. Our results show that the combination of three classifiers was able to achieve higher F-scores than the combination of two and four classifiers. Although our system was tested by using the contexts of Tweets and SMS, we believe that our system could be used with the contexts of other datasets. In future work, we are going to study other methods of ensemble learning, which we believe could be used in combination with our system for improving performance.

## References

- BIRD, S., KLEIN, E. & LOPER, E. 2009. Accessing Text Corpora and Lexical Resources. Natural Language Processing with Python. O'Reilly Media.
- CRISTIANINI, N. & SHAWE-TAYLOR, J. 2000. An introduction to support vector machines and other kernel-based learning methods, Cambridge university press.
- CUFOGLU, A., LOHI, M. & MADANI, K. 2008. Classification accuracy performance of Naive Bayesian (NB), Bayesian Networks (BN), Lazy Learning of Bayesian Rules (LBR) and Instance-Based Learner (IB1) - comparative study. International Conference on Computer Engineering & Systems (ICCES).
- ELANGO VAN, M., RAMACHANDRAN, K. I. & SUGUMARAN, V. 2010. Studies on Bayes classifier for condition monitoring of single point carbide tipped tool based on statistical and histogram features. Expert Systems with Applications, 37, 2059-2065.
- GO, A., BHAYANI, R. & HUANG, L. 2009. Twitter sentiment classification using distant supervision. CS224N Natural Language Processing, Project Report, Stanford, 1-12.
- GRYC, W. & MOILANEN, K. 2014. Leveraging Textual Sentiment Analysis with Social Network Modelling. From Text to Political Positions: Text analysis across disciplines, 55, 47.
- HOPE, L. R. & KORB, K. B. 2004. A bayesian metric for evaluating machine learning algorithms. Proceedings of the 17th Australian joint conference on Advances in Artificial Intelligence. Cairns, Australia: Springer-Verlag.
- HU, M. & LIU, B. 2004. Mining and summarizing customer reviews. Proceedings of the tenth ACM Special Interest Group on Knowledge Discovery and Data Mining (SIGKDD) international conference on Knowledge discovery and data mining. Seattle, WA, USA: ACM.
- KRIPPENDORFF, K. H. 1980. Content analysis: an introduction to its methodology, Beverly Hills, California, Calif Sage Publications.
- LIU, B. 2007. Web data mining: exploring hyperlinks, contents, and usage data, Springer.
- LIU, B. 2012. Sentiment Analysis and Opinion Mining, Morgan & Claypool.
- MARTIN-VALDIVIA, M.-T., MARTINEZ-CAMARA, E., PEREA-ORTEGA, J.-M. & URENA-LOPEZ, L. A. 2013. Sentiment polarity detection in Spanish reviews combining supervised and unsupervised approaches. Expert Systems with Applications, 40, 3934-3942.
- NIELSEN, F. Å. 2011. A new ANEW: Evaluation of a word list for sentiment analysis in microblogs. arXiv preprint arXiv:1103.2903.
- PANG, B. & LEE, L. 2008. Opinion Mining and Sentiment Analysis. Foundations and Trends in Information Retrieval, 2, 1-135.
- POLIKAR, R. 2012. Ensemble learning. Ensemble Machine Learning. Springer.
- SHAIKH, M. A., PRENDINGER, H. & MITSURU, I. 2007. Assessing Sentiment of Text by Semantic Dependency and Contextual Valence Analysis. Proceedings of the 2nd international conference on Affective Computing and Intelligent Interaction. Lisbon, Portugal: Springer-Verlag.
- TAN, S., CHENG, X., WANG, Y. & XU, H. 2009. Adapting naive bayes to domain adaptation for sentiment analysis. Advances in Information Retrieval. Springer.
- TANG, H., TAN, S. & CHENG, X. 2009. A survey on sentiment detection of reviews. Expert Systems with Applications, 36, 10760-10773.

THELWALL, M., BUCKLEY, K., PALTOGLOU, G. & CAI, D. 2010a. Sentiment strength detection in short informal text. *Journal of the American Society for Information Science and Technology*, 61, 2544-2558.

THELWALL, M., BUCKLEY, K., PALTOGLOU, G., CAI, D. & KAPPAS, A. 2010b. SentiStrength. University of Wolverhampton.

WILSON, T., HOFFMANN, P., SOMASUNDARAN, S., KESSLER, J., WIEBE, J., CHOI, Y., CARDIE, C., RILOFF, E. & PATWARDHAN, S. 2005. OpinionFinder: a system for subjectivity analysis. *Proceedings of HLT/EMNLP on Interactive Demonstrations*. Vancouver, British Columbia, Canada: Association for Computational Linguistics.

WILSON, T., KOZAREVA, Z., NAKOV, P., RITTER, A., ROSENTHAL, S. & STOYANOV, V. SemEval-2013 Task 2: Sentiment Analysis in Twitter. *Proceedings of the 7th International Workshop on Semantic Evaluation (SemEval)*, 2013. Association for Computational Linguistics.

WOLPERT, D. H. 1992. Stacked generalization. *Neural networks*, 5, 241-259.