



MSc in Computing, Business Intelligence and Data Mining stream.

MSc Research Project

Effectiveness of term weighting approaches for sparse social media text sentiment analysis

Submitted by: Mulluken Wondie, B00066150

Supervisor: Geraldine Gray

Submission date: September, 2015

Declaration

I hereby certify that this material, which I now submit for assessment on the program of study leading to the award of MSc in Computing in the Institute of Technology Blanchardstown, is entirely my own work except where otherwise stated, and has not been submitted for assessment for an academic purpose at this or any other academic institution other than in partial fulfillment of the requirements of that stated above.

Author: _____

Dated: _____

Abstract

Due to the rapid expansion of social media, an open environment is created for anybody to share opinions about anything including products, political figures or events freely. These ideas affect the products or persons involved negatively or positively. Therefore, the desire to collect and understand the opinionated content from social media in order to gain competitive advantage has grown by different parties. However, sentiment analysis on social media is challenging as texts are characterized by high degree of sparsity and dimensional space. This is due to different reasons. First, texts on social media are usually short which forces users to use standard or creative contracted forms and abbreviations. Second, users are free to write in different languages, which introduces different forms of text for the same thing. Third, the openness of social media allows users to write about completely unrelated topics. And fourth, the use of different devices like mobile causes users to make spelling errors easily. This causes poor sentiment classification performance. In this thesis, the effectiveness of term weighting feature selection methods to improve classification performance on sparse social media text was investigated. First, two sparse public Twitter datasets were selected. Next, a number of data cleansing and pre-processing techniques were performed on the selected datasets. Then a Naïve Bayes classifier was trained by applying different feature selection methods. Results were compared with other methods applied on the same dataset. The use of uni-gram and bi-gram features was also investigated. Finally, based on the empirical results obtained, a new approach composed of pre-processing and Recursive Feature Selection on a combined uni-gram and bi-gram feature sets was proposed.

Keywords: Sentiment analysis, Opinion mining, Pre-processing, Sparsity, Machine Learning, Feature Selection, Social media, WordNet, SentiWordNet, RapidMiner.

Acknowledgement

First of all, I am indebted to my supervisor, Geraldine Gray, for her constructive criticism in this work, and support and motivation in the overall program for the last two years. I would also like to thank Dr. Markus Hofmann for giving me an insight on text mining in general and sentiment analysis in particular.

Last but not least, I would like to thank my wife, Israel Yohannes, for her support and patience. I love you!

Table of Contents

Declaration.....	ii
Abstract.....	iii
Acknowledgement.....	iv
Table of Contents.....	v
List of Figures	viii
List of Tables	ix
List of Abbreviations.....	x
Chapter 1. Introduction	1
1.1 Context.....	3
1.2 Motivation	6
1.3 Challenges in social media sentiment analysis	7
1.4 Research objectives	10
1.5 Thesis structure.....	11
Chapter 2. Literature review.....	12
2.1 Introduction.....	12
2.2 Background	12
2.3 State of the art.....	13
2.3.1 Machine learning approach.....	14
2.3.2 Lexicon based approach.....	18

2.4 Discussion	20
Chapter 3. Data representation and dimensionality reduction	22
3.1 Measuring sparsity	22
3.2 Feature engineering	23
3.3 Dimensionality reduction	25
3.3.1 Feature selection approaches.....	26
3.3.2 Feature extraction technique	32
3.3.3 Semantic replacement	33
Chapter 4. Methodology.....	35
4.1 Methodology.....	35
4.2 Data sources	36
4.2.1 SemiEval-2013 dataset.....	37
4.2.2 Sanders corpus.....	39
Chapter 5. Implementation and Experiment.....	40
5.1 SemiEval-2013	40
5.1.1 Data preparation and cleansing	40
5.1.2 Document representation and feature engineering	48
5.1.3 Feature extraction	54
5.1.4 Training.....	56
5.2 Sanders dataset	57
5.6 Selected approach	60

Chapter 6. Analysis of results	63
6.1 Discussion	63
6.2 Conclusion.....	65
6.3 Future work	66
Bibliography	68

List of Figures

Figure 1. Monthly active subscriptions of digital medias in millions in January, 2015.....	4
Figure 2. Mobile Internet traffic percentage per year	5
Figure 3. Example execution of REF-SVM	32
Figure 4. Methodology, adapted from (Anjaria and Guddeti, 2014)	36
Figure 5. Term Frequency distribution of SemiEval-2013 dataset	41
Figure 6. Word cloud of SemiEval-2013	47
Figure 7. Initial RapidMiner setup	48
Figure 8. Sparsity graph of data cleansing and pre-processing results.....	50
Figure 9. F-measure graph of data cleansing and pre-processing results	51
Figure 10. RapidMiner setup for feature selection methods.....	52
Figure 11. Distribution of the word “can’t” per class label	55
Figure 12. Distribution of the phrase “can’t wait” per class label	56
Figure 13. Sysnset extraction for supervised classification.....	58
Figure 14. Sentiment extraction for unsupervised classification	59

List of Tables

Table 1. Term Frequency distribution of SemiEval-2013 dataset	40
Table 2. Examples of mapped emoticons	44
Table 3. Tokenization Examples	46
Table 4. Data cleansing and pre-processing results	50
Table 5. Results of first category feature selection methods.....	52
Table 6. Results of second category feature selection methods	53
Table 7. Distribution of the word ‘can’t’ per class label	55
Table 8. Distribution of the phrase “can’t wait” per class label	56

List of Abbreviations

NLP	Natural language processing
LARS	Least Angle Regression and Shrinkage
REF-SVM	Recursive Feature Elimination
BoW	Bag of words
MI	Mutual information
IG	Information Gain
POS	Part of speech
OOV	Out of vocabulary
SMS	Short message service
PCA	Principal Component Analysis
MaxEnt	Maximum Entropy
TF-IDF	Term Frequency inverse document frequency
IDF	Inverse document Frequency
TF	Term Frequency
SVM	Support vector machine

Chapter 1. Introduction

The classification of sparse social media texts like Twitter text messages into one of the polarity levels i.e. negative, positive or neutral depending on the view contained in the text is the primary subject of this research work. This process of identifying one's view or opinion about an object or entity by studying a certain text is known as sentiment analysis. Sentiment analysis is a particular kind of text mining which focuses on classifying text documents based on how good or how bad the document is (Pang et al., 2002). Ghag and Shah (2013) also defined sentiment analysis or opinion mining as the activity of classifying the polarization of documents as negative, neutral and positive. Similarly, Mazzonello et al. (2013) defined it as the process of an automatic identification of how polar a text is. In their view; a text can represent either a single word, a phrase in a document or the entire document. In general, sentiment analysis is an activity of identifying the polar inclination of the opinion contained in a certain text. The term sentiment analysis and opinion mining are used to mean the same thing by many researchers. In this thesis, these terms are also used in the same way.

In the last 15 years, the emergence of new ways of communication like social network sites, micro-blogging services and short message service platforms (SMS) have been paramount (Liu, 2012). These new communication media have allowed users to express and share their thoughts freely about anything (Agarwal et al., 2011). This has dramatically changed the way businesses operate and products are advertised (Edosomwan et al., 2011). In fact, social networking is a long phenomenon realized by means of various technologies including the telegraph and some network sites (Edosomwan et al., 2011), but the real explosion came after 2000 (Pang and Lee, 2008). It is during this decade that Facebook and Twitter were invented. The overwhelming popularity of social media has now become a rich source of information in order to know what people are thinking about a product, a political figure, an event or anything of interest (Ghag and Shah, 2013).

However, performing analysis on texts retrieved from social media have become a more difficult task for professionals engaged in such an activity than traditional text documents like news articles. Twitter and SMS messages are short and characterized by full of irregular tokens, new contraction of words, spelling errors, completely new words, hyperlinks, and platform specific symbols like '#' in Twitter (Anjaria and Guddeti, 2014). This causes the data to be sparse and makes the classification exercise difficult.

On the other hand, alleviating these problems in order to develop an effective opinion classification approach on micro-blogs and other similar social media platforms has been given attention by the research community only recently. In the last few years, different researchers proposed various solutions from different angles, yet there is no consensus on an optimal solution.

Therefore, this thesis focuses on the identification of sentiments on social media data by overcoming the problems mentioned above specifically data sparsity. The main objective of this research is to investigate techniques that are efficient on sparse social media text classification.

Opinion mining can be at a document level, or a sentence level, or a phrase level, or a word level. The focus of this thesis is at a document level. Document level opinion classification is considered to be more difficult than phrase level classification as a document may contain mixed sentiments. For example, the tweet "I like galaxy note, but I hate galaxy tab" consists of two mixed sentiments and can be labelled differently depending on the study being conducted (Saif et al., 2012).

Techniques to solve the problem mentioned above involves different areas including statistics, machine learning, linguistics, and so on. Considering the amount of time and the number of tasks to be performed, this thesis is limited to sentiment identification of sparse social media data. Special emphasis is given to the effectiveness of term weighting approaches as dimensionality reduction

techniques to minimize sparsity and improve classification accuracy. This is performed on two public datasets namely SemiEval-2013 and Sanders.

This chapter is organized as follows. In section 1.1, the context on which this research is built is presented. Then section 1.2 describes the motivation to undertake this research, while the challenges faced by social media sentiment analysis is presented in section 1.3. Finally, research objectives and research question are given in section 1.5.

1.1 Context

Social media is an online platform that is open for users to share opinions to other users who use the same platform. Usually users have to subscribe to use the services provided by these systems. Most social media also serve as microblogs. Microblogs are different from other conventional media primarily due to the number of characters allowed in a text. For example, Twitter allows only 140 characters in a tweet and Facebook allows only 420 characters in a status update (Gautam and Yadav, 2014).

There are many social networking sites. Six Degrees and Asian avenue are regarded as some of the pioneers which were started before 2000 (Edosomwan et al., 2011), although they did not mature very well. Consequently, these and other initiatives have played an important role for the expansion of many well-known social media sites in later years. Among some of the famous social media sites, Facebook, Twitter, QZONE, Pinterest, Google Plus+, and Tumblr can be mentioned (see Figure 1). However, Facebook and Twitter are the two most widely used social media sites that serve as microblogs as well via status update in case of Facebook and tweets in case of Twitter (Pak and Paroubek, 2010).

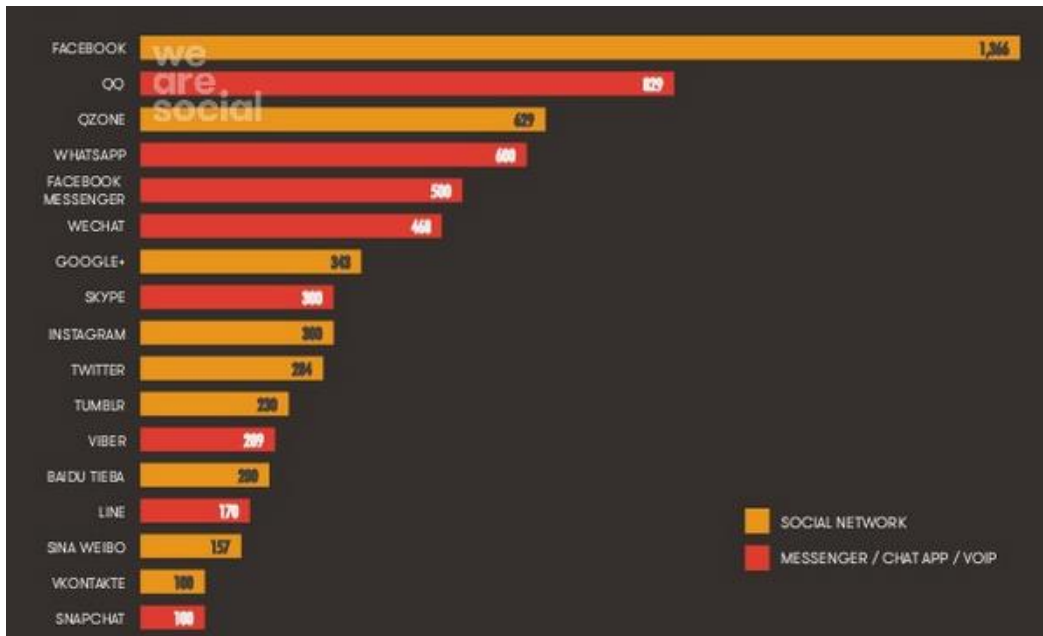


Figure 1. Monthly active subscriptions of digital media in millions in January, 2015¹

From a very recent figure taken in January 2015 presented in Figure 1, Facebook has about 1.4 billion monthly active users and Twitter has about 300 million monthly active users. It is described from the “digital, social, and mobile report”² that 700 Facebook posts and 600 Twitter messages are posted and made available at each second. Facebook allows users (above the age of 13) to register a profile. Once they create a profile, they will be able to connect with friends, send and receive messages, post status updates about themselves, comment on friend’s posts, get notified about their friend’s activities and so on. Users are also able to create or join groups and exchange ideas about shared interests.

Immediately after its creation in 2006, Twitter was embraced by many users very fast because of two reasons (Edosomwan et al., 2011). First, it serves as a microblog service and second, famous people started using it. In Twitter, a registered user can follow other user in order to see the messages of that user. The user can also re-post other’s messages called re-tweet.

¹ <http://www.slideshare.net/wearesocialsg/digital-social-mobile-in-2015>

² <http://www.slideshare.net/wearesocialsg/digital-social-mobile-in-2015>

These days, social media is the main means of communication. One of the reasons of the increase in usage of social media is the rise of mobile usage. A recent survey given in Figure 2 below shows that 33.4% of the internet traffic is made by mobile.

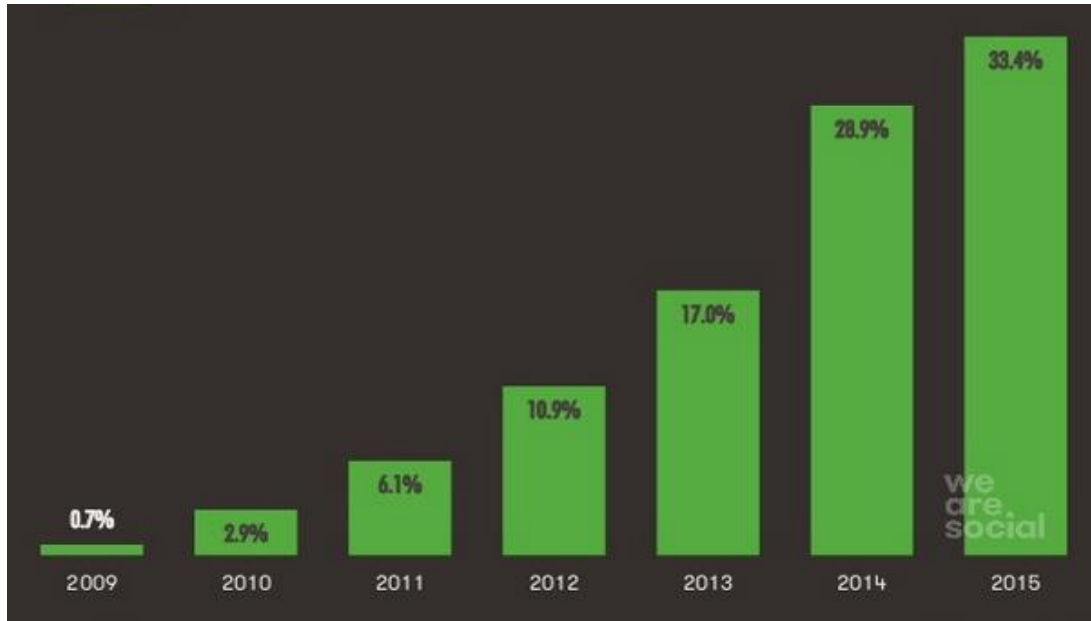


Figure 2. Mobile Internet traffic percentage per year³

A latest research has also indicated that social media is a heavily used form of advertising (Edosomwan et al., 2011). The massive magnitude of information available in social media is the main reason why many researchers consider it as an ideal source of data for sentiment mining.

The most important characteristic of sentiments that distinguish from other information containing facts is that they carry a subjective view of a person(s). Therefore, it is essential to study a pool of texts from as many individuals as possible instead of analyzing a single statement of an individual since that only denotes the idea of one person, which is not enough for sentiment analysis (Liu, 2012).

³ <http://www.slideshare.net/wearesocialsg/digital-social-mobile-in-2015>

1.2 Motivation

The rapid rise of social media, coupled with the simplicity nature microblogging functionalities, rapidly altered the way many people live enabling to share and express ideas and feelings or sentiments. The open and extremely interconnected nature of social media also enable people to look for and provide support to one another (Kiritchenko et al., 2014). Due to this, social media has become a very good source of opinionated information that could be useful to different parties including business enterprises and their competitors, customers and so on. Therefore, close supervision and analysis of sentiments from social media gives a tremendous prospect for government and private institutions.

The private sector has seen many instances where its products and services are influenced by gossips and negative attitudes posted and exchanged among people on social media.

Understanding this phenomenon, enterprises have recognized that identifying and analyzing public sentiments from social media leads to having healthier relationship with their clients, more improved knowledge of their clients' interests and faster reaction to fluctuations in the marketplace (Medhat et al., 2014).

In the general public, latest researches demonstrated that there is a very strong relationship between actions on social media and the results of many political matters (Kashyap Popat, 2013). A famous illustration commonly mentioned now a days is the use of Twitter and Facebook to shape rallies and establish unity at the time of the Arab Spring. This phenomena has affected Algeria and Tunisia, and later Egypt and Syria (Mazzonello et al., 2013). This was evident when the number of Twitter posts about politics in Egypt raised by a factor of 10 just 7 days before the then president of the country resigned (Veeraselvi and Saranya, 2014). Syria has also witnessed a sharp rise in information from Facebook by the government's opponent parties. Election is another instance where by social media is playing an important role. Barack Obama and Premier of India, Narendra Modi, beat their rivals in their respective elections partly because they used social media wisely. The last two British elections were also typical examples. Party leaders intensively used social media to accelerate their campaign and number of actions on Twitter

were found to be a reliable indicator of approval of parties and party leaders (Medhat et al., 2014). In general, Social Network sites have been extensively utilized to reflect sentiments on the web by means of texts and pictures. From the most popular network sites, Twitter has attracted many researchers in critical domains like forecasting outcomes of election, branding approval, film acceptance, trending in stock exchange, approval rate of famous individuals like celebrities and so on. In other words, opinion mining on Twitter provides a quick and effective means of identifying the sentiment of the people (Anjaria and Guddeti, 2014).

However, social media sentiment analysis is a challenging task due to different problems. Social media is much more noisy now (Medhat et al., 2014), which makes the data sparse and sentiment analysis more difficult than the conventional text classification activity. Therefore, the main motivation of this thesis comes from the desire to remove or minimize the problems caused by these challenges like sparsity for maximum possible sentiment classification performance.

1.3 Challenges in social media sentiment analysis

Extracting sentiment from social media data such as Twitter and Facebook is more challenging than traditional textual documents due to different reasons most of which are the characteristics of social media itself. One of the problems is the short length of documents from social media (Gautam and Yadav, 2014). For example, a tweet from Twitter cannot be more than 140 characters and a status update from Facebook cannot be more than 420 characters. As the number of users of social media rapidly increased in recent years due to the extensive services available including networking with peers (Jotheeswaran et al., 2012), the volume and dimension of data has also increased rapidly (Tang and Liu, 2014). Therefore, the huge volume and high dimensional data causes a problem for text classification learning activities introducing problems such as curse of dimensionality and lack of scaling ability (Tang and Liu, 2014). The problems are interrelated and well described in (Schowe, 2011) as follows.

- Social media generates a huge volume of data introducing computational complexity.
- As the number of instances in the training data increases, the number of features/dimensions required to comprehensively represent the data becomes big.
- As the number of dimensions increase, variability will be high and stability will be less.
- High variability causes sparsity which is bad for classification performance.

This shows that data sparsity is a major problem for social media sentiment analysis. However, there are only limited works that tried to address this challenge directly. As this thesis aims to address the problem of data sparsity, it is essential to have a clear understanding of data sparsity and what causes it. Data sparsity is a curse for studies involving linguistics, NLP and sentiment analysis (Kashyap Popat, 2013). Attributes that appear in the test data set but were not observed from the training data set negatively affects the performance of NLP and sentiment analysis tasks (Kashyap Popat, 2013). The problem of data sparsity is not limited to missing elements of training sets. It is common in social media training documents to find an element that appears only in one or two documents. Such an occurrence also causes data sparsity. Data sparsity in social media is caused by features of social media. Therefore, it is necessary to understand those characteristics.

➤ **The short length of texts**

It is stated earlier that social media such as Facebook and Twitter allow users to post only limited number of characters. As described in (Nakov et al., 2013), Twitter texts and SMS messages resemble more like headlines than documents. Because of this limitation posed by social media services, users tend to abbreviate and replace words with acronyms (Agarwal et al., 2011). The short and fast streaming nature of texts on social media also causes many users to spell words wrongly by mistake (Mazzonello et al., 2013). Besides the use of emoticons and various

symbols that can have different interpretations have been used extensively (Hu et al., 2013). This is one of the reasons for data sparsity problem.

➤ **Language variation**

Social media allows people from different background to comment about the same concept/topic using different languages (Alsaffar and Omar, 2014). Not only is the language different, but also there is a variation in usage within the same language. Slangs and new terms are common. Because of the informal nature of social media text, different forms of acronyms and inventive spelling is used. This causes some elements to appear in one or two documents, hence causing data sparsity.

➤ **Open social environment**

The open nature of social media allows people to write anything they want about any topic using any device (Jotheeswaran et al., 2012). In addition to introducing different connotations about the same topic, this causes to mix different topics. Therefore, a single document may contain concepts about different topics. When such a document is included in a training set of a specific topic related document, the elements stating another topic will cause sparsity. The open environment has also introduced OOV's which are another feature of social media that cause sparsity (Nakov et al., 2013). Hash tags (e.g. @ashaLadawn) and usernames (e.g. @RedArmy49) are typical OOV terms in Twitter. The symbol 'RT' to represent retweets is another OOV word common in Twitter.

➤ **Use of different devices**

The use of different devices also causes a problem. For example, people tend to use more abbreviations and are susceptible for more errors when using mobile than a computer.

1.4 Research objectives

Opinion mining in sparse social media data being the main task of this study, a number of steps have to be performed to achieve the main goal. These steps are undertaken to achieve their own sub-objectives which pave the way towards the overall objective. In general, the objectives in this study can be described in four categories as follows.

➤ Assessment of data cleansing and text pre-processing methods:

Investigation of different data cleansing and pre-processing approaches, individually or in combination with other methods, to define an efficient feature set by means of various techniques is one of the primary focuses of this study. Effectiveness of using a lexicon for feature representation or building features only from the text documents, replacement or removal of specific terms, use of stop words, stemming, normalization and emoticon mapping are some of the sub topics to be investigated. The main purpose here is to decide whether a particular combination of data cleansing and pre-processing methods improve outcomes on the text documents used.

➤ Assessment of feature reduction methods:

In this study, different feature reduction methods in order to identify which method or combination of methods is effective for sparse social media data is assessed. Feature selection methods, feature extraction technique, and semantic replacement with the help of a lexicon are also investigated.

➤ New method based on term weighting for social media sentiment analysis:

Here, a new approach is proposed by studying different term weighting approaches. Special focus is given for feature selection methods that are effective for multi-dimensional space. Some of these methods are developed for genetic data. The objective here is therefore to introduce a novel approach for social media sentiment analysis using these methods.

➤ **Assessment of the proposed approach on two sparse Twitter data sets:**

Using two sparse social media data, the proposed approach is tested to determine how effective it is compared to other methods.

With this understanding, the research question can be shaped and formed as follows:

“Can term weighting feature selection approaches improve sentiment analysis performance on social media text by reducing sparsity?”

1.5 Thesis structure

The rest of this thesis is structured as follows. Chapter 2 presents a detailed review of previous works related to this research work. In chapter 3, feature representation and feature reduction approaches are discussed and chapter 4 presents the methodology adopted. In chapter 5, a description of the experimental setup and the results of the different approaches are given. Finally, chapter 6 presents analysis and conclusive remarks.

Chapter 2. Literature review

2.1 Introduction

In this chapter, a literature review of sentiment analysis is provided. The different approaches used by different researchers are presented. One of these approaches is coming up with a score of negative and positive words and classifying the text document to the higher score. This is basically dependent on dictionary (lexicon) and is referred as lexical based approach. Another major approach is machine learning approach where a classifier is trained based on features extracted from the training documents or other external sources.

As the aforementioned techniques are the most widely applied approaches, this chapter is organized in these two perspectives. Special attention is given to contemporary researches performed on sentiment analysis and opinion mining in order to solve/minimize the problem of data sparsity so that better classification performance is obtained.

The rest of this chapter is organized as follows. Section 2.3 presents some background and section 2.3 reviews the state of the art related to machine learning approaches and lexical based approaches. Finally, section 2.4 concludes with a brief discussion.

2.2 Background

Study of sentiment analysis began only in the 90's (Liu, 2012). Previous studies were done from psychological and linguistic point of view. However, since work has started, it has drawn attention from many researchers. Liu (2012) identified two main reasons for this. Firstly, it can be applied in many domain areas. Sentiment analysis has been adopted by different fast growing commercial businesses. This has encouraged researchers towards sentiment analysis. Secondly, it has given the opportunity to work with many challenges which have

not been investigated. Today, sentiment analysis is one of the most active research areas.

Before explaining previous works in sentiment analysis, it is important to explain some problems that make comparison of results of different works of sentiment analysis difficult. Similar learners applied by different people might give dissimilar results. This is because text categorization depends on various factors including data cleaning and preparation, pre-processing, stemming, exclusion of highly frequent terms and so on. In addition, there are various options for representing documents as a set of bi-grams or uni-grams, use of lexical dictionary, part of speech usage, grammar handling and many more. A little difference in one or more of these factors may result in dissimilar outcome even though the same learning algorithm is used.

2.3 State of the art

Two approaches are used by researchers to handle text classification tasks in general and sentiment analysis in particular, Lexical based approach and machine learning approach. In each approach, different techniques including data cleansing, pre-processing, feature selection, semantic replacement, sentiment scoring and so on are used to improve classification performance. While some of the techniques are specific, some of them can be used in both approaches. For example, data cleansing and pre-processing are useful for both lexical based and machine learning approaches. However, feature selection is specifically important to machine learning approaches whereas sentiment scoring of features is only applicable to lexical based approaches. Previous research works that directly address the challenge of sparsity are limited. But there are many studies that tackle the problem indirectly from vocabulary size and feature space reduction point of view. In this section, previous works related to this research are presented.

2.3.1 Machine learning approach

In machine learning approach, sentiment identification is simply treated as a form of text classification activity. This approach is about training a classification model from pre-labelled data set which makes it basically a supervised machine learning activity (Gautam and Yadav, 2014). In this approach, text documents have to be represented using some set of features, usually extracted from the set of documents itself but can also come from another source, and finally one of the machine learning algorithms will be applied. Pre-processing in general and choosing the right set of features in particular is an important step in this approach. Some of the previous works on machine learning approach are reviewed as follows.

Go et al. (2009) investigated different feature selection methods to understand their impact on different classification learners. Among the techniques explored, Mutual information (MI) and chi-square using Naïve Bayes learner resulted in a better performance than other feature selection methods investigated. Mutual Information gave slightly a better F-score than chi-square. They also investigated different feature representation and extraction techniques. Among Uni-gram, bi-gram and tri-gram features sets, uni-gram provided the best result. As a data cleaning technique; “@user” is replaced by “USERNAME”, symbols repeated for more than two times are replaced by only two instances, and retweets are discarded. Based on these results, they have developed a framework and applied this framework on a scraped dataset using twitter API. However, their framework excluded emoticons as they argued that emoticons are simply noises and are not good enough to determine the correct polarity of a tweet. This contrasts with opinions from other researchers. For example, Pak and Paroubek (2010) argued that the sentiment of the whole document can be determined by an emoticon present in the text and terms found in the document are associated with the emoticon. This is because a text in social media platforms is limited, often only one sentence. Veeraselvi and Saranya (2014) also emphasized the importance of emoticons. In their study, they mapped certain emoticons to either of the polarities

(positive or negative) while removing vague symbols and found an encouraging result.

In another study, Pak and Paroubek (2010) emphasized on document representation as an important element for sentiment classification. They argued that a document should be represented by a vector of the presence or absence of the constituent words. The motivation for this was that the frequent occurrence of a term in a text does not necessarily say anything about its polarity. It is interesting to see that Pang et al. (2002) also used the same approach of binary term representation and got a better result. In their approach Pak and Paroubek (2010), filter methods to remove URLs and user names were applied. In order to represent documents, tokens were created by dividing the document based on space and punctuation. Further, common terms were discarded and uni-grams were constructed. In an effort to improve accuracy, a filter selection method was applied based on Shannon's entropy computation. Even though they argued that uni-grams are better to represent documents, they introduced bi-grams to incorporate the effect of negation i.e. when a negation appeared in a text, a bi-gram was formed by combining the negative indicator and the primary word. They applied their approach on the same dataset used by Go et al. (2009) and they reported that their method performed better. However, the recall rate was low. This indicates that another performance measure that combines recall and accuracy like F-measure should be used for better evaluation.

In an effort to incorporate emoticons, Agarwal et al. (2011) applied a different approach. The main difference of this method from other studies is that they used emoticons and WordNet to construct the feature space. This is a simplified approach compared to others. First, they built a word list of emoticons retrieved from Wikipedia and assigned a sentiment label to each. Second, they extended the word list by adding contracted social media specific symbols by looking up from available dictionaries with their translation, for example, gr8 = Great. As a pre-processing technique, URLs were substituted by ||U||, all negative indicators were substituted by "NOT" and a symbol that appeared more than three times

consecutively was substituted by three instances of the symbol, which is different from others who prefer to use two instances only, for example, Go et al. (2009). As a means of tokenization, Stanford tokenizer was applied. In addition, common words were discarded. WordNet was used to lookup the polarity of the terms in the word list prepared. Using the features from the wordlist, they trained an SVM upon a manually labelled Twitter document set. This method showed an improvement by 4% from the base line (Agarwal et al., 2011). The most important advantage of this method was that it had a good impact in reducing the vocabulary size.

Koncz and Paralic (2011) also proposed a new feature selection approach that uses values of text frequencies of terms in a specific class. Then he normalized this value against the sum of the texts in these classes. The main advantage of this approach was that it allocated a bigger weight for features that have smaller text frequency while remaining a typical behaviour of the specified class, which made it different from Information Gain (IG) feature selection method. The formula and analysis of the proposed approach can be found in (Koncz and Paralic, 2011). They applied this method on a movie review dataset and obtained a fairly good result. Their evaluation report shows that even though the accuracy obtained was less by 1% compared to IG, the performance in terms of processing speed was much better.

(Saif et al., 2014) is one of the research works that tried to address the challenge of sparsity directly. To tackle the problem, the study focused on stop words. The authors argued that eliminating stop words using pre-built stop word entry is not effective. As an alternative, they devised six different stop word removal techniques. Most of these have performed better than the baseline. However, the method called 'TF1' resulted in the best result (Saif et al., 2014). The basic idea of this method is that a word should be eliminated if it occurs only once. Using this method, they were able to reduce vocabulary size by 65%, sparsity by 0.37% while obtaining a very good accuracy.

Anjaria and Guddeti (2014) proposed an approach combining PCA and SVM as an effective technique of sentiment analysis. They employed a number of pre-

processing techniques including substitution of usernames, abbreviations and emoticon substitution, elimination of re-tweets, and elimination of URLs. One difference of text pre-processing employed here that is not investigated by other researchers is that they discarded redundant words in addition to redundant characters. This is unique. As a means of feature extraction; uni-grams, bi-grams and uni-grams and bi-grams combined were used. In the justification they provided, it was necessary to combine in order to effectively represent extremely positive terms like “splendid” and negative indicators like “barely”. This approach resulted in a relatively good result. However, they also reported that they applied this method on a second data set and performed relatively poorly. This implies that their approach looks like domain dependent.

PCA and SVM in combination was also investigated by Vinodhini and Chandrasekaran (2013). Using a product review dataset, they applied a series of pre-processing techniques and represented documents using bag of words (BoW) technique. As feature sets, uni-gram feature set was applied in this study. Upon the uni-gram vector of documents, PCA was applied in order to generate a diminished feature space. They applied SVM as a learning algorithm and the result obtained was fairly good, consistent with that of (Anjaria and Guddeti, 2014).

Another study by Saif et al. (2012), which focused on solving the problem of sparsity in sentiment analysis, indicated that semantic-topic features performed very well on the Stanford Twitter dataset. The basic idea behind this approach is to extract sentiment topics for all Twitter texts from a sentiment dictionary and to include these new features to the original attribute set. This combined feature space was then used as input for a classifier. The semantic topics were extracted by a technique called Joint Semantic Topic (JST) model from MPQA dictionary. Naïve Bayes was used to train the model and an accuracy of 86.3% was reported. Even though the classification performance was good, the dataset used to train the model is an automatically labelled dataset just by detecting emoticons. It could be wise to apply this method on another dataset for a firm conclusion.

With a similar interest of incorporating semantic features from a general purpose lexicon to a classifier, Ohana and Tierney (2009) used SentiWordNet scores to build a feature set that can be used in machine learning. Since synsets from WordNet follow part of speech, they used the Stanford Part of Speech Tagger⁴. This was applied on a movie review dataset and the reported classification performance shows that this approach raised the accuracy obtained from unsupervised classification using SentiWordnet on the same dataset by about 2%.

Emphasizing on high dimensional data, Schowe (2011) conducted an interesting study in order to come up with an effective feature selection method for high dimensional biological data that are characterized by a big feature space. He investigated simple filter methods like welch test, Mutual information; Hybrid Multivariate Methods like Minimum Redundancy Maximum Relevance (MRMR), Recursive Feature Elimination, Least Angle Regression; Ensemble-Methods; and other utility based ranking methods. Using these feature selection methods, a good performance was obtained. As part of his study, he implemented those techniques in RapidMiner as new operators. In order to assess their effectiveness, he conducted an experiment using microRNA-expression dataset with 67 examples and 302 features. He used MaxEnt, SVM and Naïve Bayes classifiers. His outcome report shows that the new operators outperformed other methods; MRMR providing the best performance. He also reported better computational efficiency.

2.3.2 Lexicon based approach

In contrast to machine learning approaches, lexicon based approaches do not require labelled datasets for training. Instead, these approaches mostly depend on a pre-built general purpose sentiment aware dictionary. A self-built specific dictionary can also be used but it is more time consuming and therefore its use is limited. A review of some of the previous works of this approach are presented as follows.

⁴ <http://nlp.stanford.edu/software/tagger.shtml>

Musto et al. (2014) applied different lexical dictionaries for the purpose of automatic sentiment identification including SentiWordnet, Wordnet-Affect and MPQA. First, they divided the text into smaller chunks using some dividing symbols in the text called cues. This process resulted in a series of small phrases. They used linguistic tokens like punctuation as a cue. Once these phrases were identified, their corresponding sentiment score was obtained from the dictionaries mentioned above. If there existed a negative indicator in the phrase, the sentiment of that phrase was reversed. The sentiment of the whole sentence was determined by the average of the sentiment score of each token extracted from that sentence. They applied this method to two public datasets namely Stanford Twitter Sentiment (STS) and SemiEval-2013 and they reported a good performance. In their evaluation, SentiWordnet was better than the other dictionaries by arguing that because it handles emphasized phrases and normalizes words. This is very much similar to the approach followed by Ohana and Tierney (2009) which was applied on a movie review dataset using SentiWordNet and a good classification performance was reported.

Singla et al. (2014) experimented the application of WordNet focusing on the reduction the negative effect of out of vocabulary (OOV) terms. They used a substitution technique of these words by synsets of the words' synonyms from the WordNet dictionary. They reported that this approach minimized the number of OOV terms in the document. This approach was reported to have a very good impact on sparsity. However, the method was prone to loss of information. In the example given, both words "(goes, going)" provided identical WordNet entity "go" (Singla et al., 2014). This could be problematic in sentiment identification.

On the other hand, Turney (2002) followed a slightly different approach. Rather than considering all words in a sentence, he focused on adjectives and adverbs. The semantic information contained in these linguistic parts (POS) i.e. adverbs and adjectives was used to identify the polarity of text documents. Using Mutual information algorithm, the orientation of the adjectives and adverbs was determined by how related they are to selected adjectives, for example, "Excellent"

for positive adjectives and “Poor” for negative adjectives. This method was applied to different application areas including movie reviews, banking, cars, and tourism. The report showed that this approach had proven itself to perform fairly well across domains. However, it was reported that other methods performed better on some of the datasets used.

2.4 Discussion

As it can be clearly seen from the literature, most of the works performed on machine learning have applied different feature selection methods. Weight based feature selection methods, feature extraction reduction methods like PCA and many other techniques have been applied. While PCA was effective in reducing the feature space and sparsity, it was not as effective as weight based feature selection methods for sentiment classification due to loss of information. Another technique studied is to use a feature space constructed from self-built dictionaries or general purpose lexicons. This technique was also effective in feature space and sparsity reduction but did not provide a very good performance since mapping text document tokens into terms from a dictionary cannot be perfect. Data cleansing and normalization were found to be common. There were, however, differences in the way data pre-processing was done. For example, Agarwal et al. (2011) replaced repeated characters with three instances where as many others including Pak and Paroubek (2010) preferred to replace with two instances only. In terms of learners, Naïve Bayes, MaxEnt and SVN seemed to be commonly used methods.

With regard to lexical based approaches, different dictionaries were used for unsupervised sentiment identification. WordNet was one of the most applied general purpose dictionary. While their effectiveness in reducing sparsity and feature space was good, the performance obtained was less than machine learning approaches. However, they tended to perform better across different domains.

In summary, approaches using weight based feature selection methods performed better.

Chapter 3. Data representation and dimensionality reduction

This chapter presents a review of techniques used in the experiment of this research. This is done in order to lay the foundation for the practical work and to give a good theoretical background so that works performed on the experiment chapter are seamless and logical.

3.1 Measuring sparsity

In order to manage something, it is important to understand the techniques of how to measure it (El Ghaoui et al., 2011). Similarly, to handle data sparsity problem, it is necessary to study the technique of measuring it in textual documents. In this thesis, sparsity measure is adopted as defined in (Saif et al., 2014). In order to define sparsity, consider a vector representation of texts in such a way that the texts are listed in rows and words existing in those texts in columns, i.e.

- $v = [v_1, v_2, v_3, \dots, v_m]$ denotes a vector of feature space v_j to every text j .
- m denotes total count of texts (tweets)
- $|V|$ denotes the size of the feature space.

Using these notations, Sparsity degree can be calculated as follows (Saif et al., 2014).

$$S_{degree} = 1 - \frac{\sum_{j=1}^m V_j}{m * |V|} \quad (1)$$

The degree of sparsity S_{degree} is a real number between 0 and 1, 0 representing a perfectly non-sparse dataset.

Alleviating the problem of sparsity is related to dimensionality and vocabulary size reduction methods. Whether a lexical based approach or a machine learning approach is followed, every step of a sentiment classification process contributes its own share in reducing sparsity and increasing classification performance. In this section, the steps are discussed with respect to reducing feature space and sparsity to improve classification performance.

3.2 Feature engineering

Feature construction is the activity of generating features with the help of expertise obtained from the subject under study in order to assist a training algorithm. Many researchers agree that, feature building is not an easy task. It needs expertise and requires time. As Andrew NG puts it, “Applied machine learning” is basically feature engineering⁵. Medhat et al. (2014) emphasized that before using learning algorithms, the sample data should be denoted by a clear set of features.

In this research, one of the widely applied methods known as bag-of-words (BoW) is used. In this method, all the words appearing in all the documents are identified which is called the vocabulary (Elkan, 2010). Ignoring the order and grammar of the words, documents are represented by the number of occurrences of each word. As such, the frequency of the words will be features. The vocabulary is built in two ways.

1. Only using the training data and using feature selection.
2. Using words from the training data and a lexical dictionary in combination.

The lexical dictionary used for this study is WordNet.

After the vocabulary is identified and documents are represented using a set of features, it is necessary to weight each token to know the importance of the term to distinguish one document from another. One common approach to weight terms in text classification is Term Frequency – Inverse Document Frequency (TF – IDF). The definition of TF-IDF as described by Mazzonello et al. (2013) is given as follows.

⁵ https://en.wikipedia.org/wiki/Feature_engineering

Assume the following representations (Mazzonello et al., 2013).

D = collection of documents d

$TF(t, d)$ = the frequency of term t in document d

$IDF(t)$ = the percentage of documents in document d in which term t appears

Using this representation, term frequency is defined as a division of the frequency of the term in the text by the length of the document.

$$TF(t, d) = \frac{|appearances\ of\ t\ in\ d|}{|terms\ in\ d|} \quad (2)$$

But this representation only is not enough to indicate the value of the word for training as documents contain stop-words that are not semantically meaningful like 'about', 'the', 'are', 'so' and so on. To put this into consideration, inverse document frequency (IDF) was formulated as given in the following equation.

$$IDF(t) = \log \frac{|D|}{\sum_{d_i \in D} \frac{|documents\ in\ d_i\ in\ which\ t\ appears|}{|documents\ in\ d_i|}} \quad (3)$$

where DF = the number of documents that contain the corresponding term.

This way, it is possible to give more significance to rarely occurring terms. The combination of the two i.e. TF and IDF gives TF - IDF value which is given as follows.

$$TFIDF(t, d) = TF(t, d) * IDF(t) \quad (4)$$

This gives a very clear weight since term frequency and inverse document frequency have been used to determine the significance of the term in the text. In other words, a balanced weight is obtained when the two are taken into account. It is known that uncommon terms weigh more in terms of IDF measure but less in terms of TF measure. On the other hand, common terms (stop-words) exhibit the

opposite characteristic. Therefore, terms with higher TF-IDF appears in some documents which provide some semantically sensible information to those documents. In summary, TF-IDF fulfils the following two conditions (Mazzonello et al., 2013).

- a) If the number of the occurrence of the word is high in a document, then the word is significant
- b) If a word occurs in lesser amount of documents, then the word is significant

In this research, TF-IDF is investigated as a means of document representation.

3.3 Dimensionality reduction

In text classification task, dimensionality reduction is very important to ease the process of representing documents and increase the performance of classification learners. In general, two approaches can be followed to reduce dimensionality. These are feature extraction and feature selection (Kim et al., 2009). Feature extraction is a mechanism of building extra features using the original ones and representing documents using these extra features. Feature selection on the other hand is the process of choosing only some of the original features. Comparative study of the two methods show that feature extraction is very effective in bringing down the number of dimensions by a large proportion, however, the process is susceptible to lose of semantic information from documents (Zhou et al., 2014). Feature selection on the other hand is easier for interpretation and it has been found to be instrumental in decreasing textual features (Zhou et al., 2014). However, it is a complicated process to build and identify good feature selection mechanism (Zhou et al., 2014).

As the primary focus of this research, the following section describes different feature selection methods including Information Gain, Chi-square, Recursive Feature elimination, Minimum Redundancy Maximum relevance and so on. In

chapter 5, they are investigated using the datasets chosen. Besides, the effectiveness of TF-IDF as a feature ranking mechanism is investigated. PCA as a feature extraction method is also highlighted. In the end, the use of WordNet is presented.

3.3.1 Feature selection approaches

As indicated above, the first action of text classification is to extract all the possible tokens and representing the documents according to the occurrence of the tokens in respective documents. These tokens are called features. However, this representation has two major problems (Alsaffar and Omar, 2014).

1. It makes the learning activity slow as it has to take into account more tokens than required
2. It negatively affects performance as the learner is forced to use these tokens

These problems can be considered as curse of dimensionality. The quality of any machine learning activity is related to how sparse the data is and how many dimensions it has. Practically, the measure of how far a vector is to another vector tend to be similar to how far it is from other vectors in high dimensional space (Zhou et al., 2014). The higher is the degree of sparsity of the dataset, the more potentially challenging is the task for the classifier.

Therefore, feature selection is necessary to increase accuracy by minimizing unnecessary features, increasing speed, and making better the generalizing ability of the classifier (Tang and Liu, 2014; Veeraselvi and Saranya, 2014).

While decreasing the sparsity it is crucial to keep as much as possible the Information contained in the original feature space. That means, there should be a balance between reducing the number of variables and keeping information intact. One of the techniques of minimizing sparsity is feature selection. Feature selection can be performed in different ways. Term weighting is one of the most

important methods to identify significance of terms (Kim et al., 2009). In this section, weight based feature selection methods are presented.

➤ Information Gain

Information gain is the measure of the magnitude of binary semantic content about classification assuming that only the existence of a variable and the respective target allocation is known. Practically, it tells about the decrease of entropy (the degree of ambiguity related to the variable). Sharma and Dey (2012) investigated information gain in comparison with other feature selection methods and presented its mathematical formulation as follows. Consider the following representation.

T = a text document to be classified

C = target class

n = the number of class labels in C (2 in case of binary classification)

P = the probability that a document in T is classified as class C_i

F = set of features $\{f_1, f_2, f_3, \dots, f_v\}$

Then entropy can be defined as the expected information to assign a text document in T to one of the target classes in C and is represented as follows.

$$\text{Info}(T) = - \sum_{j=1}^n (j) \log_2(P_j) \quad (5)$$

The fact that the base of the logarithm function is 2 indicates that data is represented in binary format (0 or 1).

Now, assuming that a document in T has to be classified using the set of features F, then T will be divided into v parts $\{T_1, T_2, T_3, \dots, T_v\}$. Using this, the magnitude of information to classify the document is represented as.

$$\text{Info}_F(T) = - \sum_{i=1}^v \frac{|T_i|}{|T|} * \text{Info}(T_i) \quad (6)$$

Where $|T_i|/|T|$ represents the significance of the i^{th} division and $|T_i|$ represents the corresponding entropy. This leads to the information gain formula over feature F as follows.

$$IG(F) = Info(T) - Info_F(T) \quad (7)$$

Using this formula, IG is used to choose the features that provide the biggest score. In other words, feature selection is done based on the ranks of features obtained using IG.

➤ Chi-square

Chi-square(χ^2) is used to measure the relationship between a term and a target label. With an assumption that the presence of an attribute is completely independent of the target, it is used to measure the deviation from the distribution anticipated (Zhou et al., 2014). In order to define chi-square, consider the following representation.

U = the number of mutual appearance of a word (d) and class c_i
V = the number of appearance of d where class c_i is absent
Y = the number of appearance of class c_i in the absence of d
Z = the number of documents where both d and class c_i are absent
N = the total number of text documents

Then χ^2 is given as follows.

$$\chi^2(d, c_j) = \frac{N * (UZ - VY)^2}{(U + Y) * (V + Z) * (U + V) * (Y + Z)} \quad (8)$$

Then the maximum of the results will be selected using the following formula.

$$\chi^2_{max}(d) = \max_j(\chi^2(d, c_j)) \quad (9)$$

The result of χ^2 is 0 if a feature or word d is absolutely independent of class c_j .

In chapter 2, it is indicated that most of the feature selection methods applied on text classification do not have the ability and robustness to improve sentiment classification performance of high dimensional sparse data. But there are a number approaches developed for genetic classification which is another domain suffering from high dimensionality. For example, Schowe (2011) extensively studied some of those approaches and applied in biological datasets. In this research, their effectiveness on sparse social media was investigated. Some of these feature selection methods are presented as follows.

➤ **Welch test**

Welch's test is a statistical mechanism to measure the distance between two means (Schowe, 2011). In text classification, it calculates the distance of each numerical feature to the class label. Its computation is given as follows.

$$\omega(x) = \frac{\bar{x}_+ - \bar{x}_-}{\sqrt{\frac{\sum i + (x_i - x_+)^2}{n_+} + \frac{\sum i + (x_i - x_-)^2}{n_-}}} \quad (10)$$

This is implemented in RapidMiner as "Weight by Welch-test-operator". Its typical usage is if the statistical measures of the two samples in comparison do not overlap.

➤ **F-Test score, Pearson correlation, and mutual information**

Another feature reduction technique is implemented by Schowe (2011) in RapidMiner as a filter which acts differently depending on the kind of feature and the target attribute. It identifies if the variables are numerical or not. If it finds a continuous variable (X) of type number, a target variable (Y) of type string and a class C; then it uses F-Test score which is given as follows.

$$F(x,y) = \frac{(m - C) \sum_c m_c (\bar{x}_c - \bar{x})^2}{(C - 1) \sum_c (m_c - 1) \sigma_c^2} \quad (11)$$

where σ_c^2 = the variance of class c
 m_c = the number of instances $c \in \{1,2,3,\dots, C\}$

This formula gives the variance among targets and the average variance within the targets. The relationship between two numeric variables is computed by Pearson correlation as follows.

$$R(x, y) = \frac{Cov(x, y)}{\sqrt{Var(x).Var(y)}} \quad (12)$$

And the detail computation is given as follows.

$$r(x, y) = \frac{\sum_i (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_i (x_i - \bar{x})^2 \sum_i (y_i - \bar{y})^2}} \quad (13)$$

In case of non-numeric features i.e. string variables, the relationship of two variables can be measured by using Mutual Information (MI) given in the following equation.

$$MI(x, y) = \sum_{l,m} P(x_l, y_m) \log_2 \frac{P(x_l, y_m)}{P(x_l)P(y_m)} \quad (14)$$

The operator in RapidMiner (Weight by Maximum Relevance) is implemented in such a way that it picks the right function described above according to the type of the variables.

➤ Minimum Redundancy Maximum Relevance

This is an extension of the above. What it does is going over the features in a series of a number of forward look steps with the help of correlation values and mutual information. It repeats the process by adding the attribute with the best F-test score based on a criteria (Q) given as follows.

$$F_{j+1} = F_j \cup \arg \max_{x \in X/F_j} Q(f) \quad (15)$$

Q can be one of the following two:-

a) The difference:-

$$Q_{MID} = \text{Relevance}(x, y) - \frac{1}{j} \sum_{x' \in F_j} \text{Redundancy}(x, x') \quad (16)$$

b) or the ratio between relevance and average pairwise redundancy of x given the already selected features x is an element of Fj :

$$Q_{MIQ} = \frac{\text{Relevance}(x, y)}{\frac{1}{j} \sum_{x' \in F_j} \text{Redundancy}(x, x')} \quad (17)$$

It is important to note that relevance and redundancy are materialized by correlation, F-test Score and Mutual information.

➤ **Recursive Feature Elimination (REF)**

Inspired by genetic data analysis, REF-SVM was formulated to avoid the problem of high dimensional space for classification (Samb et al., 2012). It works based on SVM but avoiding the drawback of SVM to improve feature selection power. SVM has a tendency of allocating a similar weight to related features. Sharing the weight among related attributes causes the weight of each attribute to be close to 0 or insignificant. This leads to the removal of all features all together (Schowe, 2011) without leaving any of them for classification. SVM-REF is developed to solve this problem. SVM-REF is a wrapper technique that produces ranks of attributes by a means of back-ward feature removal approach (Samb et al., 2012). The main idea behind it is to discard redundant, noisy and non-informative attributes and provide easier, more efficient and compressed attribute sets. Using a weighting technique, attributes are removed based on their discriminating power. This is done repeatedly by re-training SVM. Attributes are ranked according to the coefficients of the SVM result. In general, REF-SVM can be divided into four phases.

- 1) SVM is trained on the dataset
- 2) Attributes are ranked based on the weight vector obtained in #1
- 3) Remove attributes that have the lowest rank as determined in #2
- 4) Do steps 1 to 3 using only attributes not removed until a specified number of attributes k are left.

A typical REF-SVM execution with k=4 is given in the following figure (Schowe, 2011).

Rank	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1st round	x_1	x_{14}	x_5	x_{15}	x_{16}	x_9	x_{10}	x_7	x_{17}	x_{20}	x_3	x_8	x_{19}	x_{11}	x_{12}	x_4	x_2	x_{13}	x_{18}	x_6
2nd round	x_1	x_5	x_{14}	x_{16}	x_{15}	x_8	x_{19}	x_9	x_{10}	x_{17}	x_3	x_{11}	x_7	x_{20}						
3rd round	x_1	x_{14}	x_5	x_{15}	x_{16}	x_8	x_{19}	x_9	x_{10}											
4th round	x_1	x_5	x_{14}	x_{16}	x_8	x_{15}														
5th round	x_1	x_{14}	x_{16}	x_5																

Figure 3. Example execution of REF-SVM

➤ Least Angle Regression and Shrinkage

LARS is a dimension reduction technique based on linear regression (Keerthi, 2005). It has a close relationship with LASSO that uses L1 regularization for least squares problems. The main benefit of LARS is to rank attributes based on how important the attributes are in relation to the target variable at lower computing expense (Keerthi, 2005). LARS follows an iterative process by adding attributes that have a higher correlation coefficient against the class label at each step (Schowe, 2011).

3.3.2 Feature extraction technique

In this research, Principal Component Analysis (PCA) is used as a feature extraction technique. PCA is a feature reduction method based on orthogonal projection to map a collection of attributes characterized by a certain correlation to another collection of attributes with less or no correlation among them (Jotheeswaran et al., 2012). In other words, PCA converts an N by f matrix M to N

by t matrix O , where t is less than f . The new set of attributes are called principal components. The new principal components are less in number than the attributes used to generate them. The projection is systematic in that the 1st PCA contains the biggest possible variability; the second PCA will be accounted for the next biggest variability being constrained by the fact that it has to be orthogonal to the 1st PCA and so on (Jotheeswaran et al., 2012). The final output will, therefore, be a set of unrelated attributes which are orthogonal.

PCA is regarded as an effective feature reduction method in terms of reducing the vocabulary size and sparsity. However, it is computationally expensive and it is prone to loss of information (Anjaria and Guddeti, 2014). In this study, PCA is investigated to see how effective it is in sentiment analysis classification for social media text.

3.3.3 Semantic replacement

In this research, WordNet⁶ is used as a lexical resource. WordNet is an English general purpose dictionary developed by Princeton University and open for anyone to use. Initially, it was motivated to help information manipulation from linguistic point of view. Words like nouns, adjectives, adverbs and verbs are organized in a cluster of synsets (Veeraselvi and Saranya, 2014). These synsets in the same cluster are synonymous but at the same time being conceptually different. There is lexically and semantically established linkage among synsets. What makes WordNet more usable for NLP is its structural convenience. WordNet associates terms not only by what they mean but also based on the sense they provide.

In this thesis, WordNet is used in two ways.

1. to assist the feature construction process for machine learning
2. for unsupervised classification with SentiWordNet

⁶ Extend with <https://wordnet.princeton.edu/>

SentiWordNet is a list of words taken out of the WordNet dictionary that carry sentimental information and is open for researchers (Ohana and Tierney, 2009). SentiWordNet is useful to undertake unsupervised sentiment analysis as it provides words and their corresponding sentiments.

Chapter 4. Methodology

4.1 Methodology

This thesis follows a slightly different methodology than commonly used in data mining because of the difference exhibited by sentiment analysis compared to conventional data mining. The main subject of sentiment mining is opinion or view which makes it different from general classification tasks. For example, if we look at conventional text classification, the task is to categorize texts into one of the predefined classes without considering the senses the texts carry. Sentiment analysis can be defined as a set of {E, A, S, U, T} where E represents the thing or event that the view is written about, A refers to the set of attributes or characteristics of E, S refers to the emotion or polarity conveyed in the view, U refers to the owner of the view, and T refers to the time the view was expressed (Mazzonello et al., 2013). Therefore, a sentiment analysis starts with the intention of identifying positive or negative sentiments (S) about a certain event, item, person, product or any object of interest (E) on a certain moment in time (T). Once that is identified, the first task is to collect relevant texts about the object of interest from relevant social media, usually Twitter, or any other source written by different users (U). If a supervised classification approach is followed, the collected text documents should be labelled as positive, neutral or negative for training. In this research, an already labelled text is used and that is the starting point. For performance reasons and technical convenience, the labelled text is stored in MSSQL server using MS Visual studio. Once the labelled corpora is stored, data cleansing, normalization and pre-processing have been performed. Next, using the Bag of words approach, the text documents are represented in the form of vectors. Features (A) are built using words extracted from text documents and the WordNet lexicon. In order to select relevant attributes for training, feature reduction is performed. Upon the selected features, a classifier is applied using RapidMiner and R. Finally, the trained model is applied on unseen dataset.

Having this in mind, the methodology followed is depicted in the figure below adopted from (Anjaria and Guddeti, 2014).

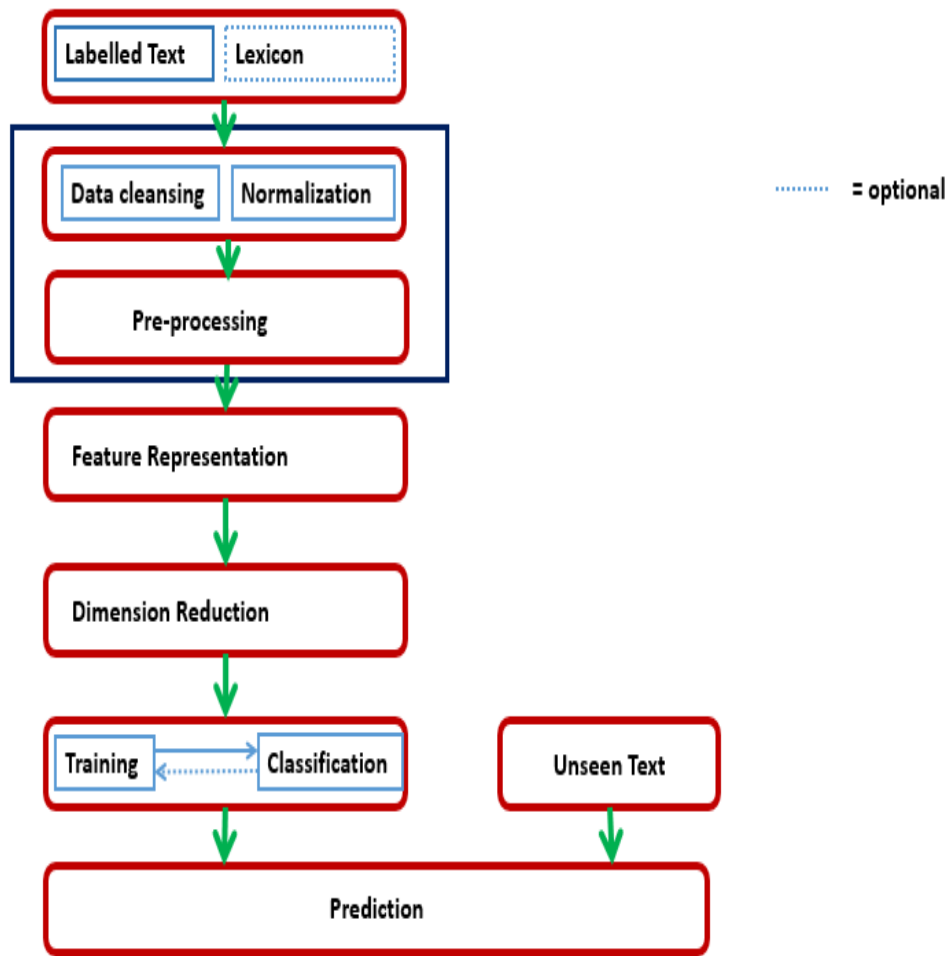


Figure 4. Methodology, adapted from (Anjaria and Guddeti, 2014)

4.2 Data sources

Choosing a good data source is critical. A self-developed crawling was used to collect social media text and manually labelling them as negative, positive or neutral took a lot of time just to complete few texts. Labelling manually for this research purpose is time intensive. Therefore, a public data set is used. Even though this research focuses on social media data, only twitter datasets are used due to a mere reason of data accessibility. At first, the Stanford Twitter Dataset was considered due to the high number of tweets. But a further investigation showed that this dataset was labelled automatically using emoticon symbols. This

puts its correctness in doubt. With the help of the review of the different datasets accessible for public research explained in (Saif et al., 2013), SemiEval-2013 and Sanders datasets have been chosen. These two datasets are annotated (labelled) manually which makes them reliable. In addition, according to the review provided in (Saif et al., 2013), they are very sparse compared to other datasets. SemiEval-2013 is the sparsest dataset out of the 9 datasets in the review. This was another motivation to choose the two datasets as this thesis focuses on improvement of classification for sparse social media data.

Before going to the detail explanation of the datasets, it is important to understand some terminologies specific to the datasets (Agarwal et al., 2011).

- Emoticons - A symbol or set of symbols used to represent opinion givers feelings in digital communication, for example ☹ = sad face
- Target user - In tweets, this is used to mention another account using “@”. This way, the holder of this account will be notified.
- Hash Tag – This is used to point to another subject using “#”. This way, someone can make her/his ideas more noticeable.

4.2.1 SemiEval-2013 dataset

This dataset is released for the purpose of competition in order to promote research in twitter sentiment mining⁷. The tweets span a period of 1 year from Jan, 2012 till Jan, 2013. For the competition, two sub-tasks were specified. One of them is sentiment identification at expression level and another is sentiment identification of the complete message. If a message has negative and positive content, the instruction states that the message should be categorised to the stronger sense. In this research, only message level sentiment analysis was taken into account. The data contains tweets about various topics including famous people like Steve Jobs, items like iphone, and activities like football.

⁷ <https://www.cs.york.ac.uk/semeval-2013/task2/index.html>

➤ Data Retrieval

Only the ID of the tweets was provided as twitter's policy does not allow to make the actual tweets available. But Python code was provided together. Using the given python code, the following data was downloaded.

- a. Development:- 1651 development datasets out of 2000 Since there were deleted messages
- b. Training:- 9116 training data out of 12000 due to the same reason in a
- c. Twitter test:- 3813 out of 4000 due to the same reason in a
- d. SMS test: - 2094 out of 4000

Notes:- the following comments were made about the dataset.

- ✓ Training and test datasets can be combined.
- ✓ There is no separate training SMS message. The reason that only test SMS data is provided is to see how well the developed system performs.
- ✓ The dataset contains positive, negative, neutral, objective, and 'objective-or-neutral'. The last three should be considered as neutral.

➤ Evaluation

The evaluation technique decided to be used as specified by the competition was the average of the F-measures of negative and positive classes. F-score is regarded as a better measure over accuracy when one of the class labels outnumbers the other. In such situations, accuracy may provide a high value based on the superior class even if the classifier performs badly on the inferior class. F-measure is given as follows⁸.

$$F - \text{score} = 2 * \frac{\text{precision} * \text{recall}}{\text{precision} + \text{recall}} \quad (18)$$

⁸ https://en.wikipedia.org/wiki/F1_score

Once the F-score of the negative and positive classes are calculated using the above formula, the overall F-score will be determined by averaging the two as follows.

$$F - \text{average} = \frac{F - \text{score}(+) + F - \text{score}(-)}{2} \quad (19)$$

4.2.2 Sanders corpus

This dataset is a manually annotated twitter dataset provided by sanders lab⁹. It has a collection of ~5500 tweets spanning from 2007 till 2011 related to topics Microsoft, Apple etc. Due to the fact that twitter's policy does not allow to distribute, they provided the IDs of the tweets and a corresponding python code to download. Even though the tweeter IDs are 5513, only 4592 were retrieved using the code provided. Other tweets appeared to be deleted. Out of this, 457 are positive, 495 are negative, 2114 are neutral and 1526 are irrelevant. Further, Irrelevant tweets were discarded.

⁹ <http://www.sananalytics.com/lab/twitter-sentiment/>

Chapter 5. Implementation and Experiment

5.1 SemiEval-2013

5.1.1 Data preparation and cleansing

The training data was built by combining the development dataset and the training data sets. This gives 11316 tweets for training. The test set which contains 3813 tweets was used for test without any change. The SMS test set was also used as a separate test set as specified by the competition.

The dataset contains a number of irregular terms and symbols. This can be considered as noise. In Twitter, there are some unique symbols in use and they make the vector space of the training environment large. The feature space of SemiEval-2013 dataset was very large. Investigation of the term frequencies in the training set showed that most of the words occurred only once in a single document. Table 1 below presents the term frequency of the SemiEval-2013 training dataset.

Table 1. Term Frequency distribution of SemiEval-2013 dataset

Category by number of occurrences	Total frequency
1-5	20553
6-10	1398
11-20	959
21-50	622
51-100	213
101-500	155
501-1000	33
>1000	22

Out of the 20553 words that occurred 1 to 5 times, 15982 of them occurred once. This indicates that a lot of document specific or user specific words were extensively used. The graph below (Figure 5) shows the frequency distribution of the feature space.

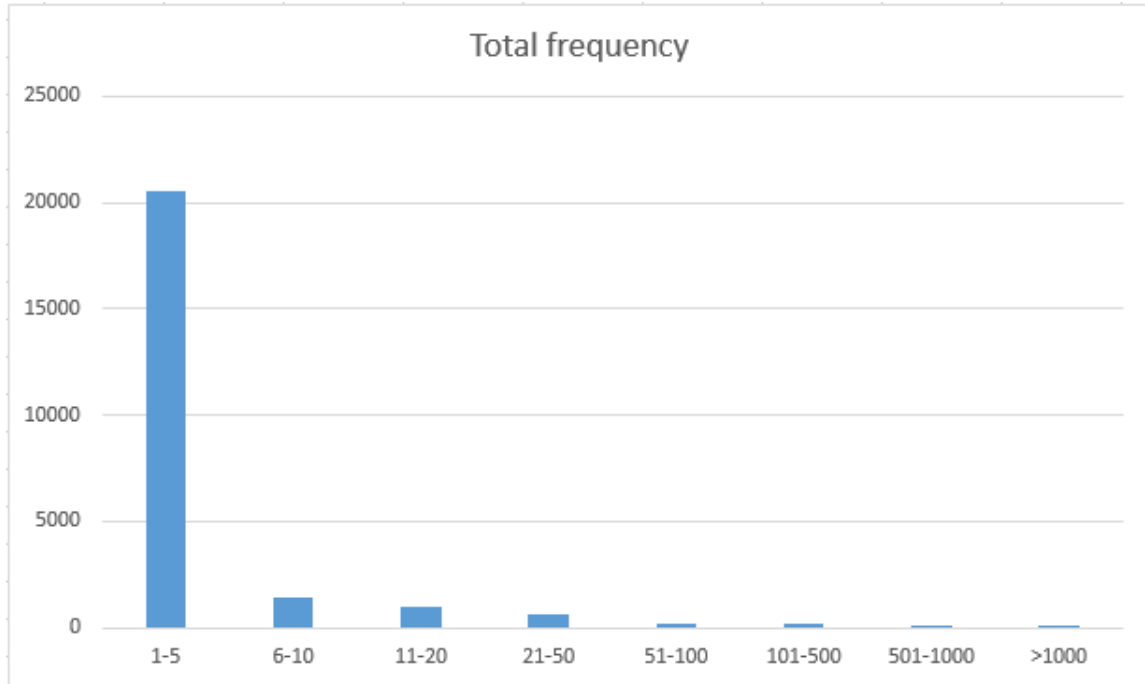


Figure 5. Term Frequency distribution of SemiEval-2013 dataset

Therefore, data cleaning is an important element of this research. In order to achieve this a number of functions have been developed at database level using T-SQL.

Data cleansing

It was noted earlier that social media data is full of irregular content (Koncz and Paralic, 2011; Veeraselvi and Saranya, 2014). For example, if we look at twitter, tweets consist of particular syntax including references or links to other sites or pages, Universal Resource Locator (URL) texts, re-tweets and so on. Such content should either be represented in a unified format or cleaned all together before the data is used by a classifier (Pang and Lee, 2008). This is a very important element of the whole process as it is useful in maintaining only relevant information in a uniform and consistent form (Han et al., 2012). Uniform and consistent form means the various forms of a term can be boiled down to a single feature. Before describing each data cleansing activities, 8000 duplicate tweets were identified in

the training dataset. Some of them were marked with the symbol 'RT' to mean retweet. All duplicate tweets were removed. Having said this, the phrase "original dataset" will be used to refer to the distinct tweets, i.e., duplicates removed. The different data cleansing techniques are described as follows.

- a. User names beginning with '@' were removed.

Username indicators which start with '@' are removed as there is no semantic information loss.

- b. URL's in the format 'http://...' and html/xml tags were removed.

In many cases, HTML character notations such as > and < are included in text documents. These elements have to be either removed if they do not have an impact on the content of the text document or replaced with their corresponding symbols i.e. ">" should be replaced with "<" for example (AlSumait et al., 2008). Other researchers' suggestion is that HTML & XML markup tags must be identified and eradicated (Musto et al., 2014). In this research, it was decided to remove such tags.

- c. If a character was repeated more than two times, then it was replaced by two instances of the character.

The rationale behind this decision is that there are many words in the English language where a character appears twice in a row like look, loop and so on. By replacing with two instances, it is possible to avoid the risk of misrepresenting these words.

For example, the word 'looooooooooveeee' becomes 'loovee'. The implementation algorithm is given as follows.

Algorithm: Removal of repeated characters

```
1: Function Removal of repeated characters(T) → The token to discard repeated characters from
  2: for each characters c ∈ {English character a-z or A-Z and a symbol in the English language} →
    3: if T contains consecutive characters c and number of c >=3 do
      4: Replace the repeated instances with two instances of the character
    5: end if
  6: end for
7: end function
```

After the implementation of this procedure, the text

Oooooooooohhhhhhhh me likey Houston +6 tomorrow. Hope the numbers line up

Became

ooh me likey Houston +6 tomorrow. Hope the numbers line up

And

I want to go watch Denzel's new movie tomorrow! Someone take meeeeeeeeeeee.

Became

I want to go watch Denzel's new movie tomorrow! Someone take mee.

- d. Hash tags beginning with '#' were discarded.

Hashtags are used to mark an important element so that the tweet can be searchable using the hashtag. However, the opinions of the individual who has written the tweet is usually contained not in the hashtag but in the other words with in the tweet. Therefore, hashtags can be safely removed without significant semantic loss of information.

- e. Single characters such as '~' that were considered to be irrelevant were removed.

Some single characters do not carry a relevant information. These characters were removed as well. Examples of removed single characters are [,], ~, {, }, -, * and so on. It is important to note here that single quote (') was not removed as it was found to be important for bi-gram feature formation in later phases. This is because, the same character is used to represent apostrophe in the English language.

f. Mapping Emoticons

The SemiEval-2013 dataset has emoticons. Since emoticons are powerful in conveying messages, a list of common emoticons together with the senses they infer is prepared and used for mapping purposes. Other emoticons which do not match to the list are removed.

Some of the common emoticons used in social media are collected and assigned a sentiment¹⁰. Some of the examples of the emoticons used are presented in Table 2 below.

Table 2. Examples of mapped emoticons

symbol	meaning	sense
:)	Happy	happy
:-)	Smile	happy
=)	Smile	happy
:D	Big Smile	happy
:-D	Big Smile	happy
:(	Sad	sad
:-('(	Sad	sad
:(	Crying	sad

¹⁰ <https://code.google.com/p/emoticony/wiki/Emoticons>

The application of the tasks mentioned above on the original dataset reduced the vocabulary by 41%. This was a good achievement as it made the data simpler to represent and easier to understand.

Pre-processing

Before starting to describe the different pre-processing techniques, it is important to explain the experimental set up. The primary tool used for the classification task is RapidMiner. Based on previous research, it was understood that Naïve Bayes, SVM and Maximum Entropy performed better. However, to my knowledge, Maximum Entropy is not implemented in RapidMiner. Therefore, R was used to run Maximum Entropy. In order to proceed with pre-processing, the three algorithms were tested on the cleansed dataset using the methods given in this section. Naïve Bayes resulted in a better performance. Therefore, Naïve Bayes was used as baseline. Using this learning algorithm, different pre-processing techniques were applied as follows.

a. Tokenization:-

The process of tokenization is dividing a given document into required smaller sub elements called tokens. Mohammad et al. (2013) also defined a tokenizer as a tool separating a text into a series of tokens, which approximately match to words and punctuation marks. Many researchers in the field of Natural Language Processing agree that this is a fundamental process. Kiritchenko et al. (2014) suggested that tokenizing is even more critical for sentiment analysis or opinion mining than it is for Natural Language Processing as sentiment bearing content is sparsely and irregularly represented in unexpected format. For example, the single presence of a symbol like “😊” could state the overall sense of a sentence or phrase. Mohammad et al. (2013) underlined that activities like those listed as data cleansing in this section should be performed before tokenization. As rightly pointed out, failing to do so will affect the tokenization process and may even provide a wrong sense as part of the

html tag (!) might be interpreted as liking something or a positive sense in general. Kiritchenko et al. (2014) emphasized that there is no unique correct approach to apply tokenizing, rather the best algorithm is subject to the activity to be performed.

In this research, the RapidMiner operator “tokenize” was used by setting the mode parameter to “linguistic tokens” and the language to “English”. This means that a word in the English language will be a token. It appeared that this tokenization technique made more sense compared to others. In Table 3 below, the results of the chosen tokenization technique and another tokenization method using “non letters” is presented for understanding purpose.

Table 3. Tokenization Examples

Text	Mode	Tokens
Welp , I made it to #MLG Dallas ... even the speed limits are bigger in Texas O.o Tomorrow , Sat , Sun ... INTERVIEWS! !! keep an eye out :)	linguistic	! #MLG) : Dallas I INTERVIEWS! O.o Sat Sun Texas Tomorrow Welp an are bigger even eye in it keep limits made out speed the to
	Non letters	Dallas I INTERVIEWS MLG O Sat Sun Texas Tomorrow Welp an are bigger even eye in it keep limits made out speed the to
(2 of 3) ...And despite what you may have seen on TV , the Jersey Shore is a place beloved by (and home to) many many good people ...	linguistic	() And 2 3 Jersey Shore TV a and beloved by despite good have home is many may of on people place seen the to what you
	Non letters	And Jersey Shore TV a and beloved by despite good have home is many may of on people place seen the to what you

As shown Table 3 above, the ‘non-letters’ option excluded special characters. But such kind of characters are very common in social media data. Therefore, the “linguistic tokens” option with language ‘English’ was used.

b. Word Filtering

After extracting units produced by tokenization, the filtering step followed. Removing stop words, the ones that are short and do not convey significant meaning, decreased vocabulary size. In order to check if the word is in the stop-word list in English, RapidMiner operator “Filter stopwords” was used.

c. Stemming

Stemming is a technique that converts inflected words to their corresponding base words (Han, Shen et al. 2012). For example, the different forms of the verb "work": "working", "worked", "works" will end up as one stem "work". For this research, the porter stemmer operator of RapidMiner was used.

After data cleansing and pre-processing steps, a word cloud was generated to identify the most frequent terms in SemiEval-2013. Figure 6 below shows the word cloud generated.

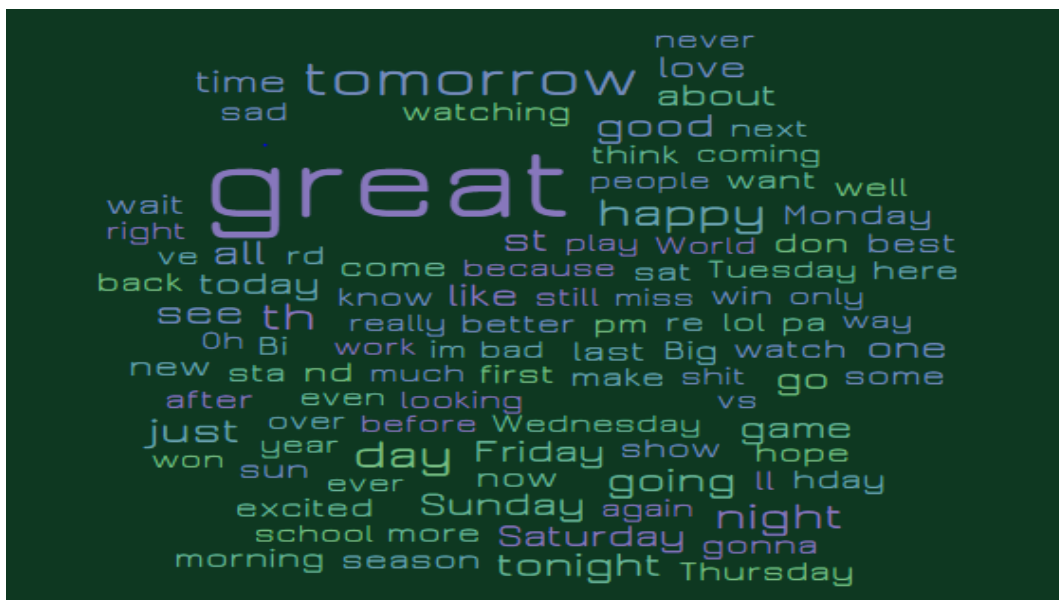


Figure 6. Word cloud of SemiEval-2013

The word cloud contains mixed words that can represent sentiments and non-sentimental words. For example, words like great, good, never, love, sad, happy bad and so on are potentially important words for sentiment analysis. On the other hand, words like tomorrow, Friday, Sunday and so on are not expected to carry a sentiment. Besides, we see a number of short forms and misspellings.

5.1.2 Document representation and feature engineering

Document representation and feature construction is a critical step in text classification activity. In this research, the Bag-of-words (BoW) method is used. Using the RapidMiner operator “Process Documents from Data”, the documents were broken into linguistic tokens. The figure below shows the initial RapidMiner set up.

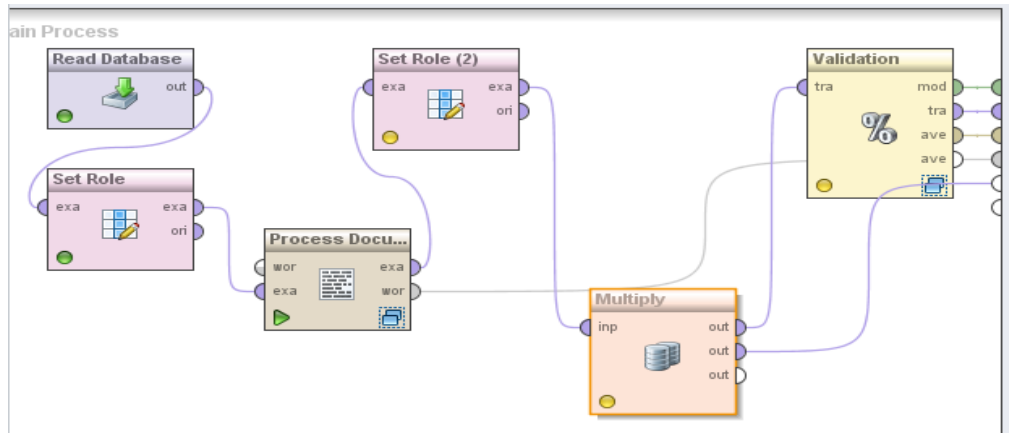


Figure 7. Initial RapidMiner setup

The purpose of each operator is given below.

<i>Read Database:-</i>	<i>Used to read the sentiment from MSSQL Server</i>
<i>Set Role:-</i>	<i>Used to set the ID field as ID role</i>
<i>Process Document:-</i>	<i>At this level, it is used to include inner operators “Tokenize” and ‘transform to lower case’. At later stages, more operators will be embedded in it including filtering, stemming etc.</i>
<i>Set Role2:-</i>	<i>Used to set sentiment as a label</i>
<i>Multiply:-</i>	<i>This is used to duplicate the example set. One set is connected to the X-validation operator to be used as input for the training algorithm. The second output is used to calculate sparsity.</i>
<i>X-Validation:-</i>	<i>Set to 10-fold cross validation. Inside this, Naïve Bayes operator is used as a learning algorithm. Besides Apply model and classification performance operators are included.</i>

Using the process “documents operator”, a vector representation of sentiment documents against the tokens was generated. At this level, the “vector creation” parameter is set to be “Binary Term Occurrences”. This was important to calculate the sparsity of the documents. However, TF-IDF was used to run models by

weighting terms and using the result as a feature. Executing this set up resulted in the following (Fpos = F-score of the positive class, Fneg = F-score of the negative class).

Accuracy:-	62.06%
Fpos:-	71.62%
Fneg:-	45.14%
F:-	58.38%
Sparsity:-	0.999999982

As a subsequent exercise, “stop words” were removed and the model was executed again providing the following results.

Accuracy:-	62.75%
Fpos:-	72.56%
Fneg:-	44.61%
F:-	58.59%
Sparsity:-	0.989917344

The result was slightly improved i.e. the sparsity degree decreased by 0.010082638, Accuracy increased by 0.69% and F-Measure increased by 0.21%. In order to see the impact of data cleansing, the code fragments that removes URL's, HashTags, and Users were included step by step and results were recorded. As a final step of data cleansing, the function that maps emoticons was included. The results of the different data cleansing activities is presented below in Table 4.

Table 4. Data cleansing and pre-processing results

Action	Accuracy	F-measure	Sparsity
Original Data	62.06%	58.37507309	0.999999982
Stopwords removed	62.75%	58.59440714	0.989917344
URL removed	62.75%	58.89452028	0.98977
HashTags and users removed	63.80%	59.61959023	0.9887492
Emoticons mapped	63.83%	59.75945929	0.988410324
Filter Tokens (<3 and >95)	59.24%	56.78143161	0.98866
Prunning with TF-IDF (<3 and >95)	61.94%	58.20026213	0.98865021
Prunning with TF-IDF (<1 and >95)	63.91%	59.82531128	0.9875023

In order to see if there was a trend in the results of the different activities, a line graph of the F-measures and the sparsity measures were generated as shown below in Figure 8 and Figure 9 respectively.

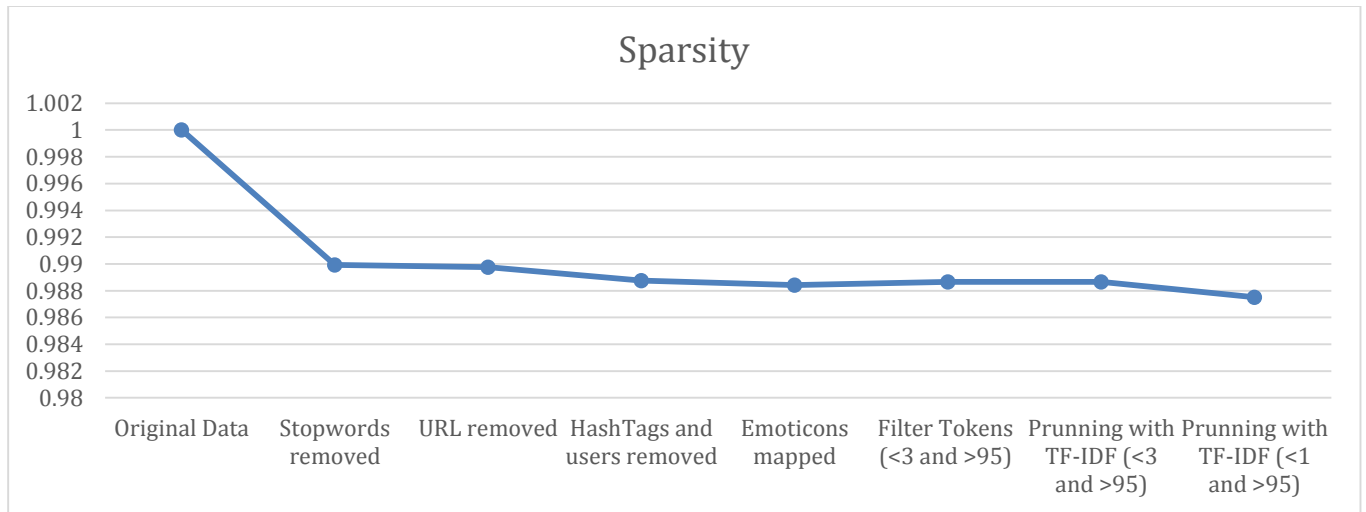


Figure 8. Sparsity graph of data cleansing and pre-processing results

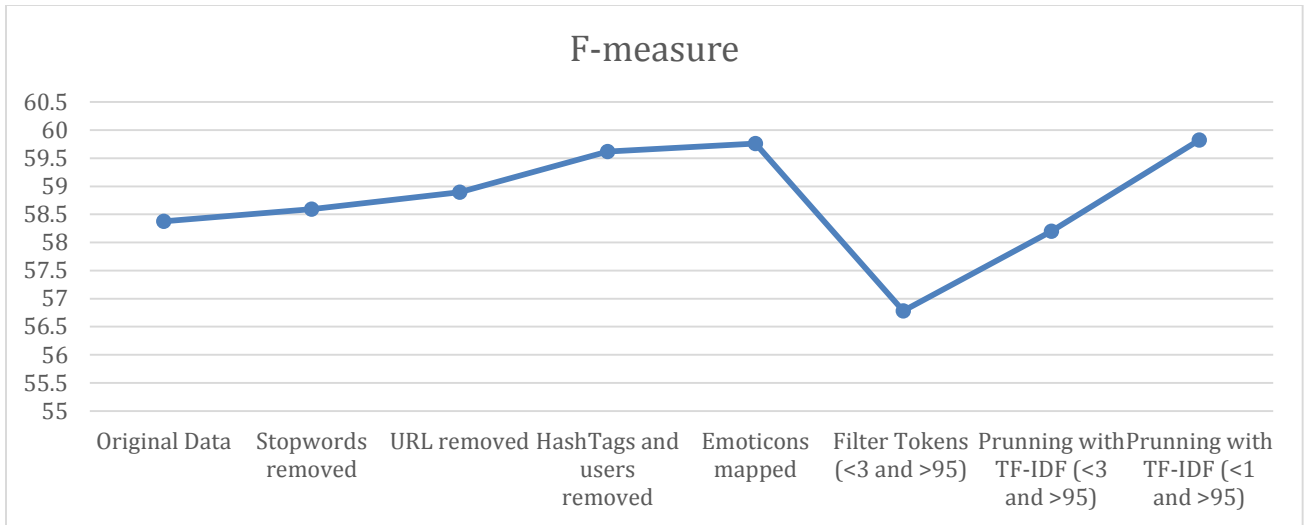


Figure 9. F-measure graph of data cleansing and pre-processing results

From the graphs (see Figures 8 and 9), the following points can be observed.

1. In general, the different data cleansing activities help to reduce sparsity
2. At the same time, those activities increased F-score value
3. There is a spike of low score in the F-score graph at Filter tokens of less than 3 characters and greater than 95 characters even though there is no significant impact on sparsity.
4. Pruning using TF-IDF with (< 3% and >95%) has raised the F-score but it is still low.
5. Pruning using TF-IDF with (< 1% and >95%) has increased the F-score value further.

From these observations, removing tokens simply based on their size of characters did not help. Besides, filtering with TF-IDF did not improve even though it was better than filtering tokens by number of characters. Stop word removal provided a good result and the other data cleansing activities also improved the result further.

Feature Selection

After pre-processing, the next step is feature selection. Using the literature as a guideline, different feature selection approaches which were thought to have

positive impact on high dimensional, sparse social media data were investigated. Even though feature selection methods are categorized as filter methods, wrapper methods and embedded methods (Schowe, 2011), for the sake of convenience for this research they were simply grouped and treated as feature selection methods used for sentiment analysis before like information gain and new to sentiment analysis but used for high dimensional data like Recursive Feature Elimination. The explanation of all the techniques applied in this section is given in chapter 3.

1. Information Gain, Gain Ratio and chi-squared

Upon the previous set up, weighting factors for Information Gain, Gain Ratio and Chi-square were applied. In this experiment, all the data cleansing and pre-processing were kept in the experiment.

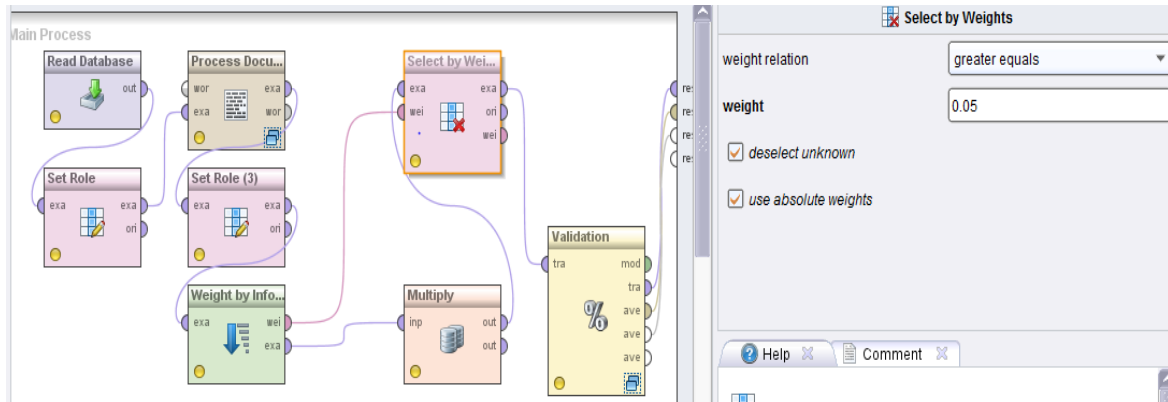


Figure 10. RapidMiner setup for feature selection methods

The weight parameter was adjusted based on the results to obtain the optimal level. The best result was obtained for Information gain at weight level 0.05. In general, lower values seemed to return better results. The results of the different filter by weighting approaches is given below in Table 5.

Table 5. Results of first category feature selection methods

Weighting method	Accuracy	F-measure	Sparsity
Information Gain	78.56%	68.58519906	0.9820502
Gain Ratio	58.92%	56.14524322	0.9835133
Chi-squared	79.48%	71.4812478	0.9811149

From the results obtained, it is clear that the overall result was improved both in terms of F-measure and sparsity. However, not all methods had a positive contribution. For example, the performance obtained from Gain Ratio was very poor. Chi-squared and Information Gain resulted in better results with chi-square resulting in the best result so far.

2. Weighting for High dimensional data

Further, feature selection methods for high dimensional data were investigated. It was made clear in chapter 2 and 3 that although these group of feature selection methods have been applied for applications with multiple dimensions, they have been barely investigated for sentiment analysis. Inspired by the positive performance they provided for dimensional data, they were applied for sentiment analysis in this research. These methods include the following (for their explanation, see chapter 3).

- Welch Test
- Maximum Relevance (MR)
- Maximum Relevance Minimum Redundancy (MRMR)
- Recursive Feature Elimination (RFE)
- Least Angle Regression (LARS)

Applying these feature selection methods, the following results were recorded as shown in Table 6.

Table 6. Results of second category feature selection methods

Weighting method	Accuracy	F-measure	Sparsity
Welch Test	67.21%	49.58427283	0.9892986
MR	75.36%	59.8533309	0.9812309
MRMR	79.86%	67.99869083	0.9810999
RFE	82.58%	75.13201066	0.9798878

The results show that some of the feature selection methods improved the results. However, some of the methods performed even worse in terms of F-measure. For example, welch test performed the worst from all feature selection methods applied. Further investigation showed that, the negative F-measure was the lowest (19%). It seems that class imbalance had a negative impact on some of the feature selection methods. In this category, Recursive Feature Elimination method resulted in the best outcome. F-measure of 75.13 was higher than any result obtained and 0.9798878 sparsity degree was lower than any result obtained so far in this research.

5.1.3 Feature extraction

Using the feature selection method that resulted in the best result i.e. “Recursive Feature Elimination”, uni-grams, bi-grams and tri-grams were investigated. The result that was obtained by combining uni-grams and bi-grams gave the best result. The accuracy went up by 3.85% to **86.43%**, the F-score went up to **81.49852419** and sparsity decreased to **0.97598576**. This has been the best result achieved in this experiment.

A closer look at the original text data showed that there was quite a big number of the word “can’t”. Out of the total 11316 training documents, 749 of the documents contained the word “can’t”. However, it seemed that this word alone did not have a discriminating power as it turned out that 299 of the documents with the word “can’t” were found to be positive and 296 were found to be negative. In other words, the negative and positive classes were almost the same. But an interesting pattern was observed taking the bi-gram “Can’t wait”. This bi-gram appeared in 118 of the documents. Out of this only 5 were negative and the rest were positive which provided a very good discriminating power. This might be the reason that incorporating bi-grams improved performance. The distribution of the word “Can’t” and the phrase “Can’t wait” is given in tabular and graphical formats as shown below.

Distribution of “Can’t”

Table 7. Distribution of the word ‘can’t’ per class label

Sentiment	#docs
negative	296
neutral	93
objective	15
objective-OR-neutral	46
positive	299
Grand Total	749

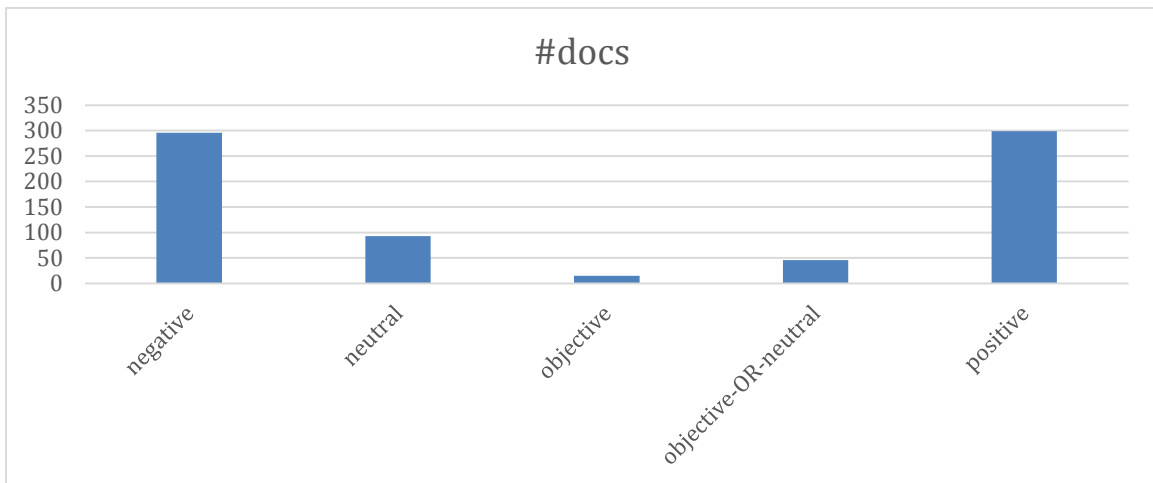


Figure 11. Distribution of the word “can’t” per class label

Distribution of “Can’t wait”

Table 8. Distribution of the phrase “can’t wait” per class label

Sentiment	#docs
negative	5
positive	113
Grand Total	118

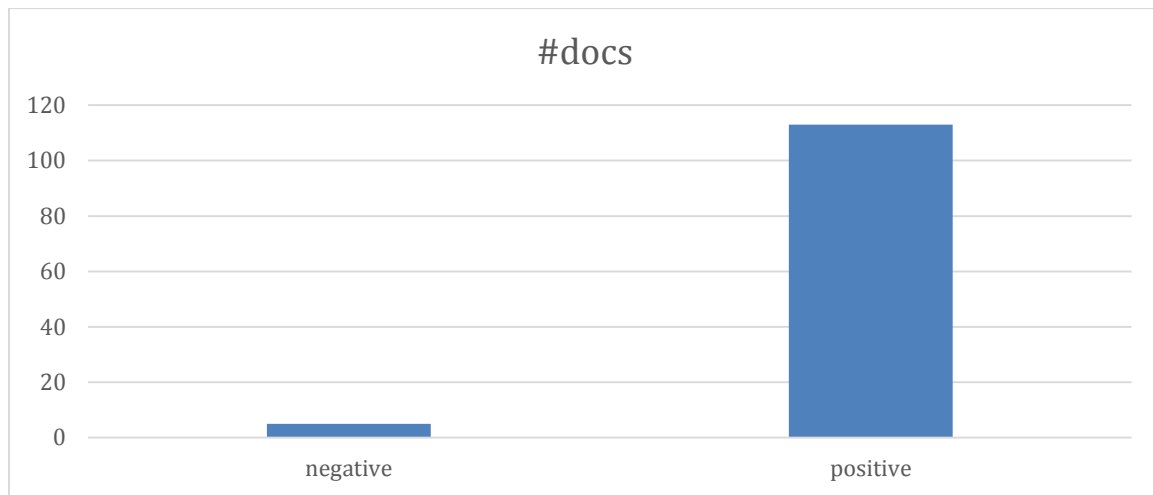


Figure 12. Distribution of the phrase “can’t wait” per class label

5.1.4 Training

For training purpose, different learning algorithms were investigated. Naïve Bayes, SVM and MaxEnt were considered and Naïve Bayes outperformed using the original training data. Taking only negative and positive samples of the training set, Naïve Bayes resulted in 62.06% accuracy, SVM resulted in 59.12% accuracy and MaxEnt returned only 60.56% accuracy. Therefore, Naïve Bayes was used as a training algorithm for subsequent investigations. It may be possible that applying the rest of the classifiers on the cleansed and pre-processed dataset provides a better outcome. However, since the main goal of this research is to investigate term weighting approaches, only the classifier that performed well on the original dataset i.e. Naïve was used.

5.2 Sanders dataset

On this dataset, the data cleansing and pre-processing techniques developed and used for the SemiEval-2013 dataset was applied. With a similar set up that resulted in the best accuracy and F-score for the SemiEval-2013 dataset, i.e. Naïve Bayes with recursive feature elimination as a feature reduction method, was executed and resulted in 86.23% accuracy and 85.94% F-measure. This was fairly a good result.

Further on this dataset, PCA as a feature reduction method was investigated. Different parameters for component numbers and weight were tested. Even though it had a very good impact in minimizing sparsity, accuracy and F-measure drastically went down. For example, at component number = 300 and weight ≥ 0.12 , sparsity became 0.98988, accuracy became 65.55% and F-measure became 61.57%.

Another method investigated was the application of a lexical dictionary. In this research, the WordNet dictionary was used first as a component of supervised machine learning approach and second with SentiWordNet as an unsupervised learning method to automatically extract sentiments.

In the first approach, synsets were extracted with the help of the WordNet dictionary using the “Find Synonyms” operator of RapidMiner as presented in Figure 13 below. Stemming was performed using the stem WordNet operator.

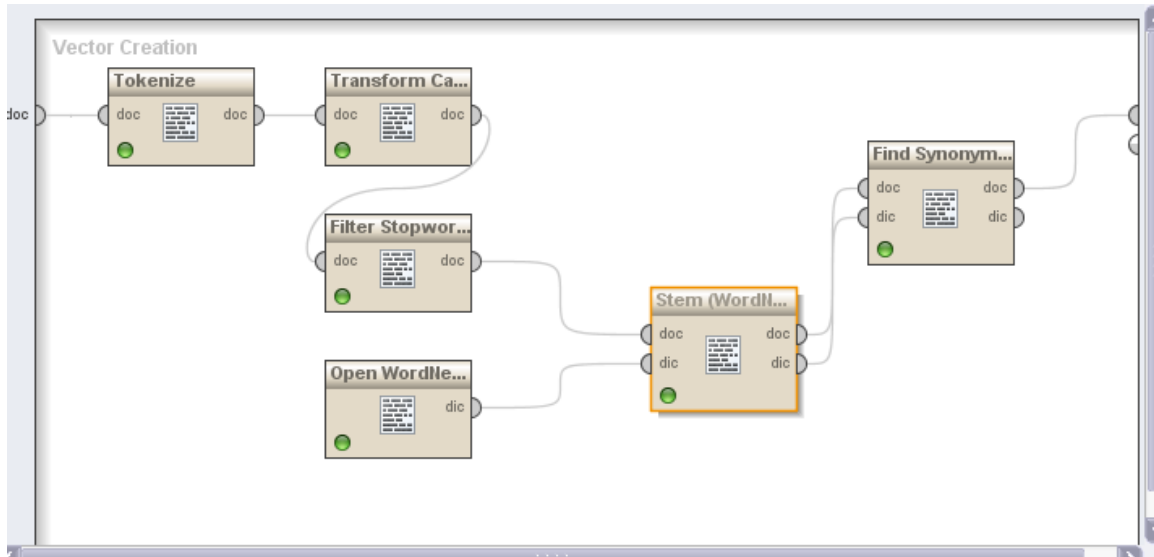


Figure 13. Sysnset extraction for supervised classification

For example, the following synsets were generated for the token “china”.

syn:China/People's_Republic_of_China/mainland_China/Communist_China/Red_China/PRC/Cathay

And the following were generated for the token “URL”.

syn:URL/uniform_resource_locator/universal_resource_locator

Continuing the training process, these synsets were used as input features for the training algorithm. The outcome of this experiment is presented as follows.

Accuracy:-	71.54%
F-measure:-	71.5312648
Sparsity:-	0.965071

In the second approach, SentiWordnet was used to extract senses. The setup follows a slightly different approach. Inside the process document operator, the WordNet stemming operator was used instead of the stem porter operator. It uses the WordNet dictionary. The doc output of the stemming operator was then

connected to the extract sentiments operator. This operator is also dependent on the WordNet dictionary. The “Extract sentiment” operator then generated the sentiment of the document as the average score of the sentiment of each token in the document. The following figure shows the RapidMiner set up of the experiment.

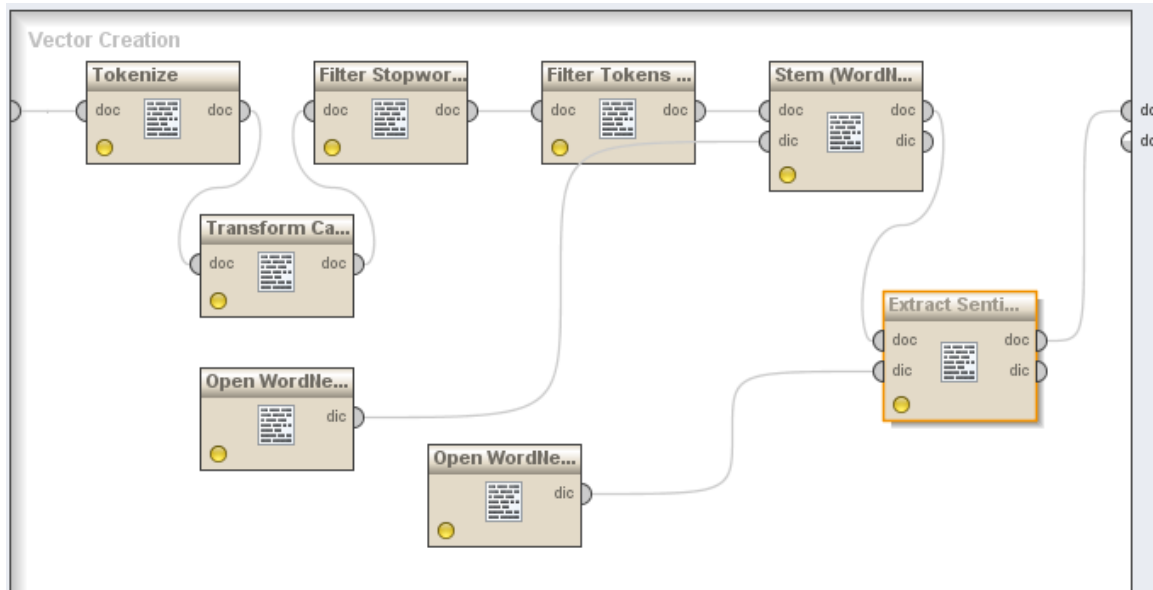


Figure 14. Sentiment extraction for unsupervised classification

This process resulted in an accuracy of 57.87% which was lower than most of the previous approaches.

From the results obtained, using SentiWordNet to extract sentiments as an automatic classifier did not provide a better result. The reason could be that dictionaries like WordNet are built for general purpose. For example, the token “ur” is used by many social media users to mean “your”. But, searching for this token in WordNet dictionary resulted in the following as a synonym.

Ur:- 1. Ur -- (an ancient city of Sumer located on a former channel of the Euphrates River)

This indicated that dictionaries developed for multiple purposes might not be effective for classifying domain specific documents. Rather extracting features using lexical dictionary and using that output as input for training algorithms seemed to be better in comparison to SentiWordNet even though it performed worse than feature selection approaches.

5.6 Selected approach

According to the experimental results, a supervised classification method based on weight based feature selection technique was proposed. The proposed approach was structured as follows.

1. Data preparation and cleansing: - this step included the activities listed in section 5.1.1.
2. Feature representation: - Bag-of-words method was used to represent documents in the form of a vector. Uni-grams and bi-grams combined were found to perform better.
3. Feature reduction: - According to the experimental results, feature selection methods performed better in general. However, Recursive Feature Elimination performed consistently better than other methods investigated. Therefore, feature selection method with Recursive Feature Elimination (RFE) method was proposed.
4. Training: - Naïve Bayes performed better on the original data. Therefore, this classifier was adopted for this research. However, these techniques can be used with other classifiers as well.

The proposed approach was tested on two test datasets. The first was a Twitter test dataset. This dataset had 3082 records out of which 1258 were positive, 454 were negative and 1370 were neutral. The Twitter test dataset exhibited the same characteristics to the SemiEval-2013 training dataset. Another dataset used to test this approach was an SMS test dataset. This dataset followed different morphological approaches. Texts were even shorter, there were more contracted forms and there was no use of hashtags and username indicators like that of Twitter. This dataset had 2094 records from which 394 were negative, 492 were

positive and 1208 were neutral. The results of the proposed model on the two test datasets are given as follows.

Results on the Twitter test dataset

True positive	1116
False positive	310
True negative	224
False negative	61
Positive precision	0.782609
Positive recall	0.948173
Negative precision	0.785965
Negative recall	0.419476
F-measure (Positive class)	0.857472
F-measure (Negative class)	0.547009
F-measure (Overall)	0.70224

Results on the SMS Test dataset

True positive	450
False positive	214
True negative	127
False negative	95
Positive precision	0.677711
Positive recall	0.825688
Negative precision	0.572072
Negative recall	0.372434
F-measure (Positive class)	0.744417
F-measure (negative class)	0.451155
F-measure (overall)	0.597786

Even though the results were lower than the result obtained using a 10-fold cross validation during training, it turned out that they were better than results obtained from previous approaches. Based on results published in (Nakov et al., 2013), the

best result obtained on this dataset so far was an F-score of 69%, and the second best score was 64.9%.

Comparing the two test sets, the model performed less on the SMS dataset. This could be because of the nature of the dataset. This model was developed using a Twitter training dataset. Therefore, all the data cleansing and pre-processing techniques were designed to fit the Twitter dataset.

Chapter 6. Analysis of results

In this chapter, an evaluation of this research work against the objectives set in chapter 1 is provided. A discussion and analysis of whether the objectives are met and how they are met is presented in section 6.1. In section 6.2, a direction for future work which could improve the process is highlighted.

6.1 Discussion

This thesis started by setting 4 objectives and a research question that were presented in section 1.5. Here, evaluation and comparative analysis between these objectives and the work performed in this thesis is discussed. The results achieved and the effect of the approaches performed against the research question are scrutinized.

➤ **Assessment of Data cleansing and text pre-processing methods:**

Extensive data cleansing was performed to remove noisy data. Usernames were removed, hashtags were removed, URL's and other html tags were removed, repeated characters that appear consecutively more than two times were replaced with two of its instances, and single characters that were considered to be irrelevant were eliminated. This brought down the vocabulary size to only 59% of the original feature space. Emoticons were also mapped. Using Bag-of-words method, documents were represented as a vector. TF-IDF was used to weight features. Tokenization, word filtering and Stemming were used and all have a positive impact in minimizing sparsity and improving classification performance. From feature extraction perspective, uni-grams and bi-grams combined have raised the classification performance. This was very similar to the methods used by Anjaria and Guddeti (2014), explained in chapter 2.

➤ **Assessment of feature reduction methods:**

With regard to feature reduction, three different techniques including feature selection, feature extraction and semantic replacement were investigated.

Information Gain, Maximum Relevance Minimum Redundancy, and Recursive Feature Elimination were some of the methods investigated as feature selection methods. As feature extraction, PCA was used to determine its effectiveness in feature reduction. Using WordNet, semantic replacement was investigated.

The results proved that even though PCA and semantic replacement were good in reducing vocabulary size, their effect on classification performance was not good as they were susceptible to information loss. It was evident from the experiment that optimal balance was obtained using Feature selection methods. Recursive Feature Elimination performed better than other methods used in this thesis.

➤ **New method based on term weighting for social media Sentiment Analysis:**

Depending the empirical results obtained, a new approach for sentiment analysis on social media text was proposed. This approach was believed to be effective if performed as follows.

1. Data preparation and cleansing (as described in section 5.1.1)
2. Feature representation (as described in 5.1.2)
3. Recursive Feature Elimination as feature selection method
4. Naïve Bayes as a classifier with uni-gram and bi-gram features combined

➤ **Assessment of the proposed approach on two sparse data sets:**

After training the proposed method with the SemiEval-2013 dataset, it was applied on two test datasets. The first was a Twitter test dataset, which was similar in character with the training dataset but had never been used as part of the training data. The second was an SMS dataset, which was different from the training dataset in character and language usage or Morphology.

The results obtained based on F-measure were 70.23% and 59.77% on the Twitter dataset and SMS test dataset respectively. This was better than results obtained using other approaches so far. As any class balancing technique to train the model

was not used, it might be possible that there could be a fair chance to improve the performance of this approach using class balancing methods.

This answers the research question that was set in chapter 1 which is **“Can term weighting feature selection approaches improve sentiment analysis performance on social media text by reducing sparsity?”**.

As explained in chapter 5, it was possible to find a good result by using feature selection techniques. These techniques were effective providing a balance between sparsity and minimal loss of information as it was proved by good classification performance. In order to be exhaustive, unsupervised classification approach was also applied using SentiWordNet on the same dataset. The results obtained were lower than most of the machine learning approaches investigated.

6.2 Conclusion

In this research, the effectiveness of term weighting approaches for dimensionality reduction and improvement of classification on sparse social media sentiment analysis at document level was investigated. Social media text is full of irregular terms introducing sparsity which makes opinion mining a difficult task. The problem was approached from different angles.

First, the state of the art to understand available techniques to solve the problem was reviewed. Presently, sentiment analysis on social media is a new but active area of research. Therefore, some researchers studied the problem and tried to provide various solutions among which two approaches were applied namely lexical based and machine learning. In this research, the pros and cons of each method were investigated.

Second, two public Twitter datasets were chosen for experiment. The choice of the datasets was influenced by the characteristics of the data. Although labelling was believed to be important to train a classifier in case of supervised approaches, due to time constraint, manually labelling a self-developed crawled dataset was

considered to be out of scope. Therefore, already labelled public datasets were chosen. Besides, in line with this research, very sparse datasets were selected.

Third, different classifiers on the chosen datasets applying various pre-processing and dimensionality reduction methods were trained. The results obtained from each method were compared. Lexical based, unsupervised approaches were also tried. Again, the results obtained with results from other methods were compared. For better evaluation, F-score to measure performance was adopted. The degree of sparsity and performance at each step when necessary was measured. Based on the results obtained, a new approach using term weighting feature reduction technique of Recursive Feature Elimination method was proposed.

Finally, this approach on unseen Twitter test dataset was tested. This method was also tested on a new SMS test dataset which was very different from tweets in terms language and syntax. The result confirmed that this approach was better compared to other methods applied on this dataset so far.

6.3 Future work

There are a number issues that were not touched in this research due to time constraint that are believed to improve the classification performance. For example, all the models were tried using highly imbalanced dataset. This may have negatively affected the result and it is possible that class balancing techniques may improve performance and therefore they are worth investigating.

Another point to note is the way how to handle negations. In this research, a specific negation handling technique was not implemented. However, the application of bi-grams was partly instrumental in this regard. Having said that, it might be important to consider effective negation handling methods.

It was also evident that social media is full of spelling errors. If an effective spelling correction was implemented, there might be a chance that it could reduce sparsity and improve classification performance.

In addition, sarcasm and irony, like any other communication, are part of social media texts. In this research, emoticons were simply mapped to certain general positive and negative bearing terms. However, there might be a high chance that some symbols were used to represent sarcasm. In this case, they could have an effect of reversing the semantic meaning of the whole sentence. Effective sarcasm handling and irony detection techniques could improve classification performance.

Bibliography

- AGARWAL, A., XIE, B., VOVSHA, I., RAMBOW, O. & PASSONNEAU, R. Year. Sentiment analysis of twitter data. *In: Proceedings of the Workshop on Languages in Social Media*, 2011. Association for Computational Linguistics, 30-38.
- ALSAFFAR, A. & OMAR, N. Year. Study on feature selection and machine learning algorithms for Malay sentiment classification. *In: Information Technology and Multimedia (ICIMU)*, 2014 International Conference on, 2014. IEEE, 270-275.
- ALSUMAIT, L., BARBARÁ, D. & DOMENICONI, C. Year. On-line lda: Adaptive topic models for mining text streams with applications to topic detection and tracking. *In: Data Mining*, 2008. ICDM'08. Eighth IEEE International Conference on, 2008. IEEE, 3-12.
- ANJARIA, M. & GUDDATI, R. M. R. Year. Influence factor based opinion mining of twitter data using supervised learning. *In: Communication Systems and Networks (COMSNETS)*, 2014 Sixth International Conference on, 2014. IEEE, 1-8.
- EDOSOMWAN, S., PRAKASAN, S. K., KOUAME, D., WATSON, J. & SEYMOUR, T. 2011. The history of social media and its impact on business. *Journal of Applied Management and entrepreneurship*, 16, 79-91.
- EL GHAOU, L., LI, G.-C., DUONG, V.-A., PHAM, V., SRIVASTAVA, A. N. & BHADURI, K. Year. Sparse Machine Learning Methods for Understanding Large Text Corpora. *In: CIDU*, 2011. Citeseer, 159-173.
- ELKAN, C. 2010. Text mining and topic models. *Lecture notes*.
- GAUTAM, G. & YADAV, D. Year. Sentiment analysis of twitter data using machine learning approaches and semantic analysis. *In: Contemporary Computing (IC3)*, 2014 Seventh International Conference on, 2014. IEEE, 437-442.
- GHAG, K. & SHAH, K. Year. Comparative analysis of the techniques for Sentiment Analysis. *In: Advances in Technology and Engineering (ICATE)*, 2013 International Conference on, 2013. IEEE, 1-7.
- GO, A., BHAYANI, R. & HUANG, L. 2009. Twitter sentiment classification using distant supervision. *CS224N Project Report, Stanford*, 1, 12.
- HAN, P., SHEN, S., WANG, D. & LIU, Y. Year. The influence of word normalization in English document clustering. *In: Computer Science and Automation Engineering (CSAE)*, 2012 IEEE International Conference on, 2012. IEEE, 116-120.
- HU, X., TANG, J., GAO, H. & LIU, H. Year. Unsupervised sentiment analysis with emotional signals. *In: Proceedings of the 22nd international conference on World Wide Web*, 2013. International World Wide Web Conferences Steering Committee, 607-618.
- JOTHEESWARAN, J., LOGANATHAN, R. & MADHU SUDHANAN, B. 2012. Feature Reduction using Principal Component Analysis for Opinion

- Mining. *International Journal of Computer Science and Telecommunications*, 3, 118-121.
- KASHYAP POPAT, B. A. 2013. The haves and the have-nots: Leveraging unlabelled corpora for sentiment analysis.
- KEERTHI, S. S. Year. Generalized LARS as an effective feature selection tool for text classification with SVMs. *In: Proceedings of the 22nd international conference on Machine learning*, 2005. ACM, 417-424.
- KIM, J., LI, J.-J. & LEE, J.-H. Year. Discovering the discriminative views: measuring term weights for sentiment analysis. *In: 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*, 2009. Association for Computational Linguistics, 253-261.
- KIRITCHENKO, S., ZHU, X. & MOHAMMAD, S. M. 2014. Sentiment analysis of short informal texts. *Journal of Artificial Intelligence Research*, 723-762.
- KONCZ, P. & PARALIC, J. Year. An approach to feature selection for sentiment analysis. *In: Intelligent Engineering Systems (INES), 2011 15th IEEE International Conference on*, 2011. IEEE, 357-362.
- LIU, B. 2012. Sentiment analysis and opinion mining. *Synthesis Lectures on Human Language Technologies*, 5, 1-167.
- MAZZONELLO, V., GAGLIO, S., AUGELLO, A. & PILATO, G. Year. A Study on Classification Methods Applied to Sentiment Analysis. *In: Semantic Computing (ICSC), 2013 IEEE Seventh International Conference on*, 2013. IEEE, 426-431.
- MEDHAT, W., HASSAN, A. & KORASHY, H. 2014. Sentiment analysis algorithms and applications: A survey. *Ain Shams Engineering Journal*, 5, 1093-1113.
- MOHAMMAD, S. M., KIRITCHENKO, S. & ZHU, X. Year. NRC-Canada: Building the state-of-the-art in sentiment analysis of tweets. *In: Proceedings of the Second Joint Conference on Lexical and Computational Semantics (SEMSTAR'13)*, 2013.
- MUSTO, C., SEMERARO, G. & POLIGNANO, M. 2014. A comparison of Lexicon-based approaches for Sentiment Analysis of microblog posts. *Information Filtering and Retrieval*, 59.
- NAKOV, P., KOZAREVA, Z., RITTER, A., ROSENTHAL, S., STOYANOV, V. & WILSON, T. 2013. Semeval-2013 task 2: Sentiment analysis in twitter.
- OHANA, B. & TIERNEY, B. Year. Sentiment classification of reviews using SentiWordNet. *In: 9th. IT & T Conference*, 2009. 13.
- PAK, A. & PAROUBEK, P. Year. Twitter as a Corpus for Sentiment Analysis and Opinion Mining. *In: LREC*, 2010. 1320-1326.
- PANG, B. & LEE, L. 2008. Opinion mining and sentiment analysis. *Foundations and trends in information retrieval*, 2, 1-135.
- PANG, B., LEE, L. & VAITHYANATHAN, S. Year. Thumbs up?: sentiment classification using machine learning techniques. *In: Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*, 2002. Association for Computational Linguistics, 79-86.

- SAIF, H., FERNÁNDEZ, M., HE, Y. & ALANI, H. 2013. Evaluation datasets for twitter sentiment analysis: a survey and a new dataset, the sts-gold.
- SAIF, H., FERNÁNDEZ, M., HE, Y. & ALANI, H. 2014. On stopwords, filtering and data sparsity for sentiment analysis of Twitter.
- SAIF, H., HE, Y. & ALANI, H. Year. Alleviating data sparsity for twitter sentiment analysis. *In*, 2012. CEUR Workshop Proceedings (CEUR-WS. org).
- SAMB, M. L., CAMARA, F., NDIAYE, S., SLIMANI, Y. & ESSEGHIR, M. A. 2012. A novel RFE-SVM-based feature selection approach for classification. *International Journal of Advanced Science and Technology*, 43.
- SCHOWE, B. Year. Feature Selection for high-dimensional data with RapidMiner. *In*: Proceedings of the 2nd RapidMiner Community Meeting And Conference (RCOMM 2011), Aachen, 2011.
- SHARMA, A. & DEY, S. 2012. Performance investigation of feature selection methods and sentiment lexicons for sentiment analysis. *IJCA Special Issue on Advanced Computing and Communication Technologies for HPC Applications*, 3, 15-20.
- SINGLA, K., SACHDEVA, K., YADAV, D., BANGALORE, S. & SHARMA, D. M. 2014. Reducing the Impact of Data Sparsity in Statistical Machine Translation. *Syntax, Semantics and Structure in Statistical Translation*, 51.
- TANG, J. & LIU, H. 2014. An Unsupervised Feature Selection Framework for Social Media Data.
- TURNEY, P. D. Year. Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews. *In*: Proceedings of the 40th annual meeting on association for computational linguistics, 2002. Association for Computational Linguistics, 417-424.
- VEERASELVI, S. & SARANYA, C. Year. Semantic orientation approach for sentiment classification. *In*: Green Computing Communication and Electrical Engineering (ICGCCEE), 2014 International Conference on, 2014. IEEE, 1-6.
- VINODHINI, G. & CHANDRASEKARAN, R. 2013. Effect of feature reduction in sentiment analysis of online reviews. *Int J Adv Res Comput Eng Technol (IJARCET)*, 2, 2165-2172.
- ZHOU, X., GUO, L., WANG, T. & HU, Y. Year. Text Feature Selection Based on Class Subspace. *In*: Data Mining Workshop (ICDMW), 2014 IEEE International Conference on, 2014. IEEE, 267-273.