AUTHOR: BANDHAKAVI, A.

TITLE: Domain-specific lexicon generation for emotion detection from text.

YEAR: 2018

This work was submitted to- and approved by Robert Gordon University in partial fulfilment of the following degree:
Doctor of Philosophy, School of Computing and Digital Media

Domain-specific Lexicon Generation for Emotion Detection from Text

BANDHAKAVI, ANIL

A thesis submitted in partial fulfilment
of the requirements of Robert Gordon University
for the degree of Doctor of Philosophy

January 2018

Abstract

Emotions play a key role in effective and successful human communication. Text is popularly used on the internet and social media websites to express and share emotions, feelings and sentiments. However useful applications and services built to understand emotions from text are limited in effectiveness due to reliance on general purpose emotion lexicons that have static vocabulary and sentiment lexicons that can only interpret emotions coarsely. Thus emotion detection from text calls for methods and knowledge resources that can deal with challenges such as dynamic and informal vocabulary, domain-level variations in emotional expressions and other linguistic nuances.

In this thesis we demonstrate how labelled (e.g. blogs, news headlines) and weakly-labelled (e.g. tweets) emotional documents can be harnessed to learn word-emotion lexicons that can account for dynamic and domain-specific emotional vocabulary. We model the characteristics of real-world emotional documents to propose a generative mixture model, which iteratively estimates the language models that best describe the emotional documents using expectation maximization (EM). The proposed mixture model has the ability to model both emotionally charged words and emotion-neutral words. We then generate a word-emotion lexicon using the mixture model to quantify word-emotion associations in the form of a probability vectors. Secondly we introduce novel feature extraction methods to utilize the emotion rich knowledge being captured by our word-emotion lexicon. The extracted features are used to classify text into emotion classes using machine learning. Further we also propose hybrid text representations for emotion classification that use the knowledge of lexicon based features in conjunction with other representations such as n-grams, part-of-speech and sentiment information. Thirdly we propose two different methods which jointly use an emotion-labelled corpus of tweets and emotion-sentiment mapping proposed in psychology to learn word-level numerical quantification of sentiment strengths over a positive to negative spectrum. Finally we evaluate all the proposed methods in this thesis through a variety of emotion detection and sentiment analysis tasks on benchmark data sets covering domains from blogs to news articles to tweets and incident reports.

Declaration of Authorship

I declare that I am the sole author of this thesis and that all verbatim extracts contained in the thesis have been identified as such and all sources of information have been specifically acknowledged in the bibliography. Parts of the work presented in this thesis have appeared in the following publications:

- A. Bandhakavi, N. Wiratunga, P. Deepak and S. Massie: Generating Word-Emotion Lexicon from #Emotional Tweets. In: Proc. of 3rd Joint Conference on Lexical and Computational Semantics, *SEM (2014)
  (**Chapter 4**)

- A. Bandhakavi, N. Wiratunga, S. Massie and P. Deepak: Lexicon Generation for Emotion Detection from Text. IEEE Intelligent Systems, January/February (2017)
  (**Chapter 4**)

- A. Bandhakavi, N. Wiratunga, P. Deepak and S. Massie: Lexicon based Feature Extraction for Emotion Text Classification. Elsevier Pattern Recognition Letters on Data Mining (2016)
  (**Chapter 5**)

- A. Bandhakavi, N. Wiratunga, P. Deepak and S. Massie: Emotion-corpus guided Lexicons for Sentiment Analysis on Twitter. In: Proc. of SGAI International Conference on Artificial Intelligence, BCS SGAI (2016) (Received best student paper award in the technical stream of the conference proceedings)
  (**Chapter 6**)

- A. Bandhakavi, N. Wiratunga, P. Deepak and S. Massie: Emotion-Aware Polarity Lexicons for Sentiment Analysis on Twitter. Communicated to : Expert Systems (2017)
  (**Chapter 6**)

Acknowledgements

Firstly, I would like to express my deepest appreciation to my supervisors, Dr. Nirmalie Wiratunga and Dr. Stewart Massie from RGU and Dr. Deepak Padmanabhan from Queen's university, Belfast. You have always offered me the best support and advice whenever I needed them. I am also extremely grateful to the strong and critical supervision you all have offered for my research - thank you very much.

This research benefited from a generous funding by the IDEAS research institute at RGU. I am very grateful for this. In the same light, I wish to further thank RGU for offering training, infrastructure to conduct research and also travel opportunities to conferences to showcase my research. This has helped me immensely in my development as a researcher .

I also appreciate the insightful feedbacks I received on this research by fellow students and staff members from the SIS research group and SICSA research communities, who played a role in making this research possible. I also appreciate the help and support from the administrative staff at CSDM.

I would also like to thank Sharad Khandelwal, Oleksander Bondarev and Rushi Luhar from SentiSum for their support and encouragement during my thesis writing.

Finally, I would like to express my gratitude to my family for their constant love, support, encouragement and prayers.

Contents

List of Figures

List of Tables

Abbreviations

BoW	Bag-of-Words
DSEL	Domain Specific Emotion Lexicon
ED	Emotion Detection
EM	Expectation Maximization
GEI	Graded Emotion Intensity
GEC	Graded Emotion Count
GPEL	General Purpose Emotion Lexicon
IR	Information Retrieval
LDA	Latent Dirichlet Allocation
MAP	Mean Average Precision
MRR	Mean Reciprocal Rank
ML	Machine Learning
MLE	Maximum Likelihood Estimation
NLP	Natural Language Processing
PMI	Pointwise Mutual Information
SA	Sentiment Analysis
TEC	Total Emotion Count
TEI	Total Emotion Intensity
UMM	Unigram Mixture Model
WED	Word Emotion Document

Dedicated to my dearest parents and sister

Chapter 1

Introduction

With the advent of the internet and social messaging platforms (e.g. Twitter) large volumes of user feedback data is generated on a day-to-day basis. This data is a reflection of users' daily thoughts, opinions and views about phenomena from world and political events to brands, services and consumer products. Such user feedback data forms a wealth of knowledge for businesses and enterprises to understand their customers' pain points in order to formulate strategies and action steps to avoid customer churn. Customer experience (CX) is reaching high competitive levels with the focus of the biggest brands such as *Apple*, *Amazon*, *BMW*, *John Lewis* etc on delivering exceptional customer service. This is evident from the increasing market size for companies focussed on social listening[1], feedback aggregation[2] and omni-channel analytics[3,4].

Emotion is an important factor that influences overall human behaviour which includes day-to-day tasks such as reasoning, decision making and interaction. Though emotions are subjective, they occur in objectively deducible ways in text [2]. Emotion detection concerns the computational study of natural language expressions in order to identify their associations with different emotions such as *anger, fear, joy, sadness, surprise* etc. Until recently sentiment analysis is used widely to gauge customer experience by analysing customer feedback data [3]. However such binary insights do not reveal the experience of the customers in detail. For example, a customer might feel welcomed at a restaurant, but could be unsatisfied with the service and the price. In

[1]https://www.brandwatch.com/
[2]https://uk.trustpilot.com/
[3]http://www.sentisum.com/
[4]http://www.clarabridge.com/

such a scenario a simple star rating or a binary categorization of the experience as positive or negative, is not sufficient to understand the feelings of the customer. Given that there is unprecedented access to emotion-rich content through tweets, blogs and discussion posts there is a great opportunity and need to build automatic tools, in order to understand the emotions of the users. For instance, an emotion analysis system can be developed to determine customer attitude towards products/services from review data. Such a system is very useful from the service provider's perspective, in order to track engaged as well as dissatisfied customers, and also from the customer's perspective, to gain insights about other customers' purchase experience. Systems established in this area include WordNet-Affect [4], NRC word-emotion lexicon [5] and EmoSenticNet [6]. Further, there are plethora of visual and analytical tools to detect emotion from text in the form of API services. For example, Qemotion[5] identifies emotion towards entities in social media text where as Synesketch assesses general emotion of the given text and also visualizes it.[6]

Despite the proliferation of systems already in existence, emotion detection still remains an open research field due to its ever-increasing application domains, linguistic nuances, differing contexts and interpretations across cultures making it challenging to automatically analyse a piece of text for emotion.

1.1 Related Research Fields

Emotion detection research has over the years been influenced by advances in Natural Language Processing (NLP), Sentiment Analysis (SA) and Text Classification (TC). In the sections below we highlight the relationships between emotion detection from text and each of the above mentioned research fields to understand how advances in each of these fields influenced research in emotion detection from text.

Natural Language Processing is the field of computer science concerned with the study of how computers interact with human languages. Therefore, it is highly relevant to textual emotion detection since emotion is typically expressed in an unstructured manner using text. Emotion detection can be done using some of the techniques developed in NLP such as the method of

[5] http://www.qemotion.com/
[6] http://krcadinac.com/synesketch/

splitting text into individual words (tokenization), mapping words to their root forms (lemmatization) and the process of marking-up words corresponding to particular part-of-speech (PoS tagging). These techniques are typically available from standard NLP suites such as the GATE[7] and StanfordCoreNLP[8], but they need an extension to address peculiar challenges of emotion detection particularly applied to social media that contains informal and non-standard content. It can be noted, however, that such extensions are already under-way in addition to new NLP tools developed specifically for social media platforms (e.g. TweetNLP[9]). Also, NLP draws from computational linguistics and statistics to develop rules to handle human language. Such rules are also essential for emotion detection, for instance, in lexicon generation and contextual modelling of language. However, existing NLP rules are often agnostic of the challenges to mine emotions in social media content. This is an area we explore in this thesis.

Sentiment Analysis concerns the computational study of natural language text (e.g. words, sentences and documents) in order to identify and effectively quantify its polarity (i.e positive or negative) [7]. More specifically, the main tasks of sentiment analysis comprise the extraction of opinion polarity (positive or negative), the target or specific aspects of the target to which the opinion refers to, the holder of the opinion and the time at which the opinion was expressed [8]. Sentiment lexicons are the most popular resources used for sentiment analysis, since they capture the polarity of words. These lexicons are either hand-crafted (e.g. opinion lexicon [9], General Inquirer [10] and MPQA subjectivity lexicon [11]) or generated (e.g. SentiWordNet [12] and SenticNet [13]) using linguistic resources such as WordNet [14] and ConceptNet [15]. However, on social media (e.g. Twitter), text contains special symbols resulting in non-standard spellings, punctuations and capitalization; sequence of repeating characters and emoticons for which the aforementioned lexicons have limited or no coverage. As a result there are lexicons developed to capture the domain-level informal and creative expressions used on social media to convey sentiment [16, 17]. The extraction of such lexicons is possible with limited effort, due to the abundance of weakly-labelled sentiment data on social media, obtained using emoticons [18, 19]. The work done in this thesis is inspired by lexicon-based approaches for sentiment analysis and proposes a generative word-emotion lexicon for emotion detection from text.

[7] https://gate.ac.uk/
[8] http://nlp.stanford.edu/software/corenlp.shtml
[9] http://www.cs.cmu.edu/ark/TweetNLP/index.html

Text Classification is the automatic classification of a collection of documents into a set of predefined classes. Supervised machine learning techniques such as support vector machines and naive Bayes [20] are popularly used in text classification. Emotion classification is among the most widely studied problems in emotion detection from text, where supervised machine learning methods are leveraged to classify text documents [1, 21] into emotion classes, induced from emotion theories proposed in psychology by [22], [23] and [24]. Of the two common approaches to emotion modelling, discrete emotions has been subject to extensive exploration over the continuum approach [1], [25, 26]. This is explained by the fact that in psychological research it is often easier to acquire discrete quantifications (such as a Likert scale) through user studies compared to continuous real values.

In the case of emotion detection, a supervised learning algorithm is trained on a set of emotion labelled training documents. Such documents are typically represented as vectors that lie within a space whose dimensions correspond to a sub-set of selected features[10] from the original training documents. Once the training is complete, the classifier is expected to correctly predict the class of a previously unseen test document that follows the same document-to-label distribution as the training set. A limitation of text classification in supervised learning is the need for labelled training data. However on Social media (e.g. tweets) weakly-labelled emotional data by users with emoticons and emotion hashtags is available in abundance which can be leveraged to train supervised classifiers and further be transferred to model emotions in other domains. These solutions are very useful in the context of lexicon-based emotion detection. In this thesis we explore such utility in domain-specific emotion lexicon (DSEL) induction for emotion detection and for extraction of effective features for emotion text classification.

1.2 Research Motivation

Text is an important means not just to convey facts but also to express emotions. Text-based emotion detection is the computational study of natural language expressed in text, in order to identify its association with emotions such as *anger, fear, joy, sadness, surprise* etc. Emotion

[10] typically words contained in documents

knowledge discovery can directly impact applications concerning industry (e.g.customer experience[11], employee engagement[12]), media (e.g. analysing online reactions towards political[13] and sports events) and government organisations (e.g. understanding the emotions, feelings of a community[14]). However, there are challenges involved in modelling fine-grained subjectivity and the subtlety of emotive expressions in text.

Sentiment lexicons [12] (due to lack of granular emotion information) and general purpose emotion lexicons (GPELs) [4, 5] (due to the static and formal nature) are inadequate for emotion detection in domains such as social media, where vocabulary change happens dynamically. In particular on Twitter, user generated vocabulary like emotion hashtags (e.g. #romeisawesome, #loveisbliss, #RIP) and emoticons (e.g. :-), :-(, :P etc) are found in plenty, in contrast to the formal vocabulary in GPELs (see Figure 1.1). Further the association between words and emotions vary from one domain to another. For example *Glee* may normally indicate *joy*, but, would need to be interpreted as neutral in a corpus of documents talking about the television series with the same name[15]. Therefore predetermined modelling of word-emotion associations as in GPELs and sentiment lexicons becomes limitedly effective for in-depth analysis of emotions in different domains.

The aforementioned challenges can be alleviated by learning domain specific word-emotion polarity lexicons (DSELs) which can not only capture the word-emotion associations within the domain but also quantify them. A DSEL can be deployed for a variety of tasks concerning emotion detection. In particular they offer useful knowledge to design a range of document representations from simple binary to frequency counts to more sophisticated emotion concepts. Emotions expressed by individual words can be captured using a lexicon which is very useful to fragment large pieces of text into segments that are emotion related and emotion unrelated. This kind of emotion detection is useful as a precursor to representations that are effective for emotion classification. Also DSELs quantify the association strength between words and emotions, therefore they can be used in emotion ranking tasks at word, sentence and document level. DSELs learnt on large corpora can form very useful and powerful tools for sifting through social

[11] http://www.brandembassy.com/blog/the-6-core-emotions-in-customer-experience-and-why-they-matter
[12] http://www.kanjoya.com/kanjoya-and-twitter-co-present-new-model-of-employee-engagement-an-even-better-workplace-and-a-competitive-advantage/
[13] http://www.bbc.co.uk/news/blogs-trending-32071377
[14] http://www.number27.org/wefeelfine
[15] http://en.wikipedia.org/wiki/Glee_(TV_series)

media in order to provide emotion related insights to applications, businesses and organisations. Further DSELs can be transferred to search and index vast amounts of emotional content (e.g. song lyrics, image tags/descriptions, title/comments of online videos etc) on the world wide web in order to gain insights about the emotions expressed through multimedia (e.g songs, images, videos etc) .

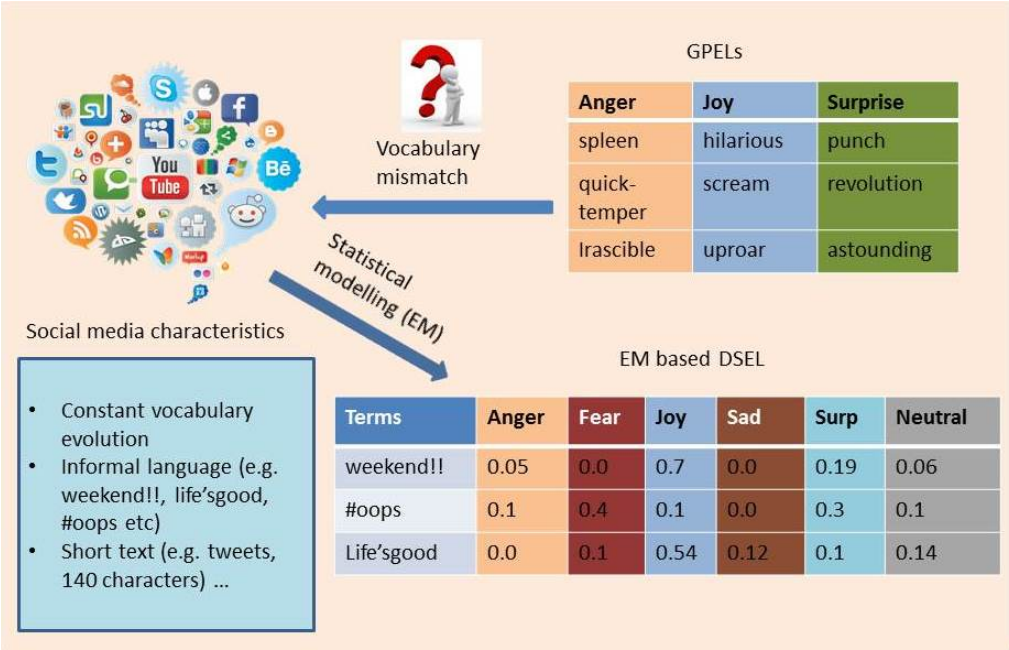


FIGURE 1.1: Motivation for learning DSELs

The central aim of this research is to develop effective tools for textual emotion detection and apply them to solve novel problems which require emotion related insights.

In order to address issues discussed above in relation to emotion detection from text, this thesis explores the following research questions:

1. How to induce a highly accurate domain-specific emotion lexicons that can quantify the emotionality and neutrality of words using a corpus of emotion-labelled documents ?

2. How to extract effective features from a domain-specific emotion lexicon for emotion text classification?

3. How can the relationship that exists between emotion and sentiment be exploited to improve performance of sentiment analysis?

1.3 Research Objectives

In this thesis, we address the problem of emotion detection from text using a generative mixture model-based emotion lexicon to jointly model the emotionality and neutrality of words. We model the problem of emotion detection with a focus on variety of tasks such as word-emotion classification, word-emotion ranking and document-emotion classification. Specifically, we address the following five objectives:

1. To develop an effective methodology to automatically generate a domain specific emotion lexicon (DSEL) to capture word level associations with emotions.
2. To utilize the knowledge of the DSEL effectively to extract lexicon based representations of text for emotion text classification using machine learning.
3. To investigate the role of hybrid text representations obtained by combining lexicon based features and non-lexicon based features such as n-grams, POS features and sentiment features for emotion text classification using machine learning.
4. To study the role of emotion knowledge for sentiment analysis on social media.
5. To comprehensively evaluate the different methods/strategies proposed for emotion detection from text and also the methods to apply emotion knowledge for sentiment analysis.

1.4 Contributions

Figure 1.2 highlights the main contributions of this thesis towards emotion detection from text. The contributions in this research are made within the framework of supervised learning for emotion detection from text. We now present the details of each contribution in this thesis.

In the first contribution a corpus of emotion-labelled documents is utilized to learn a domain-specific word-emotion lexicon. The quality of the proposed lexicon is evaluated through emotion

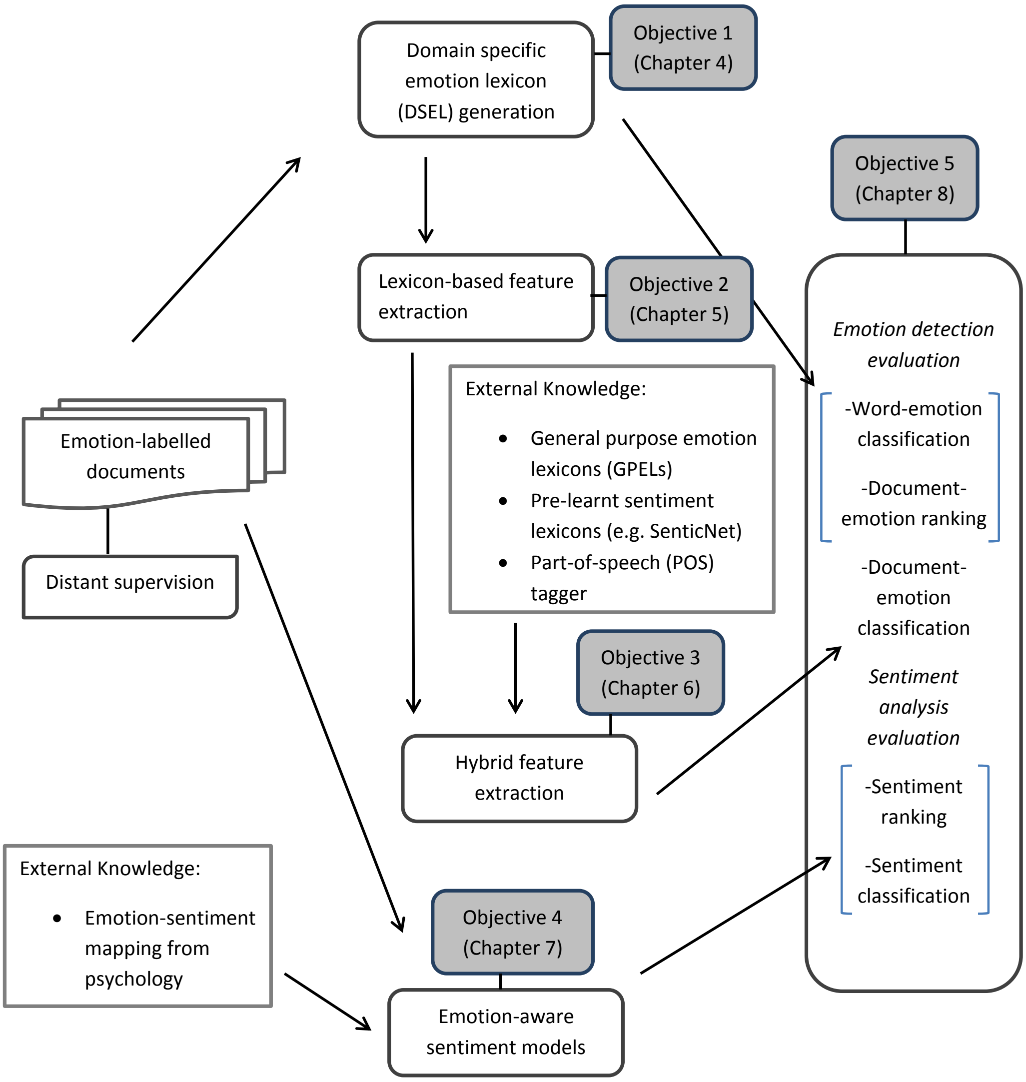


FIGURE 1.2: Objectives/Contributions of the thesis

detection tasks such as word-emotion classification and document-emotion ranking on benchmark datasets which is part of the fifth contribution. The second contribution is where the knowledge of the lexicon learnt in the previous stage is utilized to extract effective lexicon-based features for emotion classification. We evaluate the quality of these features in document-emotion classification task using machine learning on benchmark data sets. The third contribution is the extraction of hybrid features for emotion classification by combining emotion lexicon features and other standard features proposed in the literature. The fourth contribution is about the effective utilization of an emotion-labelled corpus of documents for sentiment analysis by extracting emotion-aware sentiment lexicons. Here theoretical constructs from psychology are adopted in the learning phase of the sentiment lexicons. Finally, the proposed lexicons are evaluated through sentiment analysis tasks on benchmark datasets.

The first contribution of this research is the induction of a domain-specific word-emotion lexicon from a corpus of emotion labelled documents. We propose a generative unigram mixture model (UMM) to characterise the linguistic structure of real-world emotion documents. The proposed mixture model then iteratively estimates the optimal emotion and neutral language models for the given corpus of documents using expectation maximization (EM). Thereafter the word-emotion lexicon is obtained by normalizing the language models which captures the emotionality and neutrality for each word in the form of a probability distribution. The uniqueness of the proposed lexicon lies in its ability to model both the emotionality and neutrality of words which is not possible using other automatic lexicons learnt using state-of-the-art methods such as point wise mutual information (PMI) and supervised latent Dirichl et allocation (sLDA).

A second contribution is the extraction of emotion sensitive features to represent documents for emotion text classification. We proposed several different feature extraction methods that utilize the knowledge of the proposed UMM lexicon in many different ways to extract powerful features that can effectively represent documents to discriminate their emotional orientation. The proposed features go beyond the simple word-count based lexicon features proposed in the literature which cannot model the subtle variations in the associations between words and emotions. This is very important for emotion detection, given the complex ways in which it is expressed in natural language. In this contribution the focus is entirely on the representation learning aspect of text classification. We extensively study the role of different text representations with and without the knowledge of the proposed UMM lexicon to highlight the contribution of the knowledge it captures for a machine learning classifier to learn class decision boundaries.

Our third contribution is the development of emotion-aware models for sentiment analysis. Here the knowledge of emotion-labelled documents and the emotion-sentiment mapping from psychology are combined to learn sentiment lexicons for Twitter sentiment analysis. We proposed two different ways in which such lexicons can be learnt from Twitter data. This is very useful for social media opinion mining, given the dynamic nature of its vocabulary which is very difficult to comprehend using standard sentiment lexicons which do not account for rich emotive expressions that are highly sentiment bearing.

Other contributions of this research include the exploration of hybrid representations for emotion text classification by combining the knowledge of the proposed lexicon based features and other standard features used in literature such as n-grams, sentiment lexicon features and part-of-speech (POS) features. We also conducted a detailed evaluation of the proposed methods in comparison with state-of-the-art baselines through a variety of tasks concerning emotion detection and sentiment analysis on benchmark data sets.

1.5 Thesis Overview

The rest of this thesis is outlined as follows: In Chapter 2 we present a review of literature related to emotion theories and computational approaches for emotion detection from text grouped under: keyword-based, corpus-based and machine learning. We also highlight the findings of works in literature that focus on studying the interplay between emotions and sentiment as this thesis aims to uncover their relationships, especially in social media.

In Chapter 3, we present background details about the main general purpose and domain specific emotion lexicons that form the baselines used in this research. We also provide details of the different features extracted using n-grams, sentiment lexicons and POS features which act as baselines in the evaluation of emotion text classification. We also provide details about the evaluation datasets, machine learning classifiers and the performance metrics employed.

Chapter 4 presents our generative UMM for word-emotion lexicon generation. We begin with the motivation for such a mixture model by using some real world examples, followed by the technical details of the iterative process expectation maximization used to estimate the parameters of the mixture model. The chapter concludes with a walk through example illustrating the lexicon generation on a sample data set.

In Chapter 5, we present the various feature extraction methods using the knowledge of the word-emotion lexicon proposed in Chapter 4. We follow the same procedure to extract these features from other lexicons learnt using PMI and sLDA. We also illustrate the different hybrid features extracted by combining lexicon based features and the baseline features outlined in Chapter 3. Finally we also visually explain the different lexicon based features proposed in this chapter using example data.

Chapter 6 presents the different methods proposed to learn emotion-aware sentiment lexicons for Twitter sentiment analysis. The chapter begins with the formulation of the two different methods, mathematically and visually, followed by a walk-through example illustrating the finer details on a sample data.

A comparative study of all the proposed methods for emotion detection and sentiment analysis discussed in Chapters 4, 5 and 6 together with baselines appears in Chapter 7. We evaluate the performance of the proposed methods and the baselines on a variety of tasks such as word-emotion classification, document-emotion ranking, document-emotion classification, sentiment ranking and sentiment classification. We used benchmark data sets from different domains such as blogs, tweets, news headlines and incident reports for performance evaluation in emotion detection and a wide variety of benchmark Twitter data sets for the sentiment analysis tasks. We report statistical significance observed in performance using t-test.

We conclude this thesis in Chapter 8 with a summary of our main contributions and desirable extensions for future work.

Chapter 2

Literature Review

In this chapter we first present the various emotion theories, followed by a review of state-of-the art work in emotion detection from text grouped under: *keyword-based, corpus-based and machine learning approaches*. Thereafter we review literature concerning sentiment analysis that incorporate emotion related information. We highlight previous work in the field and identify the gaps which this research seeks to address.

2.1 Emotion Theories

Emotion theories are outcomes of formal studies undertaken in the field of psychology about various emotions expressed and experienced by humans. The focus of these have been to identify the basic emotions and organize them into structures (e.g. ontologies, taxonomies). In the following sections we detail the most popular emotion theories studied in psychology. Each theory differs from the other in terms of the set of emotions identified as the basic or primary emotions. However there exists a set of emotions that are commonly identified by all the emotion models as basic or primary and it is such commonalities that we hope to exploit for computational emotion model generation.

2.1.1 Ekman Emotion Model

Paul Ekman, an American psychologist focused on identifying the most basic set of emotions that can be expressed distinctly in the form of a facial expression: *anger, fear, joy, sadness, surprise and disgust*. The key idea here is that each identified Ekman basic emotion can be discriminated from the rest by its facial expression characteristics [22].

2.1.2 Plutchik's Emotion Model

Robert Plutchik, a psychology professor emeritus at the Albert Einstein College of Medicine also proposed the concept of basic emotions. Unlike the Ekman emotion model Plutchik's emotion model defines eight basic emotions such as *anger, anticipation, disgust, joy, fear, sadness and surprise* [24]. These basic emotions are arranged as bipolar pairs namely: *joy-sadness, trust-disgust, fear-anger, surprise-anticipation*. All others are a result of combinations, mixtures, or compounds of these primary emotions.

Further according to this model, emotions differ in their degree of similarity to one another and each can exist in varying levels of intensity and arousal. Plutchik's emotion model is depicted as a three-dimensional circumplex wheel in Figure 2.1. In the figure each petal of the wheel represents an emotion with different levels of intensity, with intensity increasing. e.g *boredom* when intensified becomes *loathing*. The figure also describes the relations between emotion concepts, which are analogous to the colors on a color wheel. The eight sectors are designed to indicate that there are eight primary emotion dimensions arranged as four pairs of opposites. In the exploded model, the emotions in the blank spaces are composite emotions that are a mixture of two of the primary emotions. e.g. *love* is a composite emotion, which is a mixture of the two basic emotions *joy* and *trust*.

2.1.3 Parrot's Emotion Taxonomy

Parrot organised emotions in a three level hierarchical structure [23] corresponding to primary, secondary and tertiary emotions with *anger, fear, joy, sadness, surprise and love* representing the primary set of emotions. Unlike the Ekman and Plutchik emotion models, the emotion *love*

Primary Emotion	Secondary Emotion	Tertiary Emotion
Love	Affection	Adoration, affection, love, fondness,liking, attraction, caring, tenderness, compassion, sentimentality
	Lust	Arousal, desire, lust, passion, infatuation
Joy	Cheerfulness	Amusement, bliss, cheerfulness, gaiety, glee, jolliness, joviality, joy, delight, enjoyment, gladness, happiness, jubilation, elation, satisfaction, ecstasy, euphoria
	Zest	Enthusiasm, zeal, zest, excitement, thrill, exhilaration
	Contentment	Contentment, pleasure
	Pride	Pride, triumph
	Optimism	Eagerness, hope, optimism
Surprise	Surprise	Amazement, surprise, astonishment
	Irritation	Aggravation, irritation, agitation,annoyance, grouchiness
Sadness	Sadness	Depression, despair, hopelessness, gloom, glumness, sadness, unhappiness, grief, sorrow, woe, misery, melancholy
	Disappointment	Dismay, disappointment, displeasure
	Shame	Guilt, shame, regret, remorse
Anger	Exasperation	Exasperation, frustration
	Rage	Anger, rage, outrage, fury, wrath, hostility, ferocity, bitterness, hate, loathing, scorn, spite, vengefulness, dislike, resentment
	Disgust	Disgust, revulsion, contempt
	Suffering	Agony, suffering, hurt, anguish
Fear	Horror	Alarm, shock, fear, fright, horror, terror, panic, hysteria, mortification
	Nervousness	Anxiety, nervousness, tenseness, uneasiness, apprehension, worry, distress, dread

TABLE 2.1: Parrot's Emotion Taxonomy

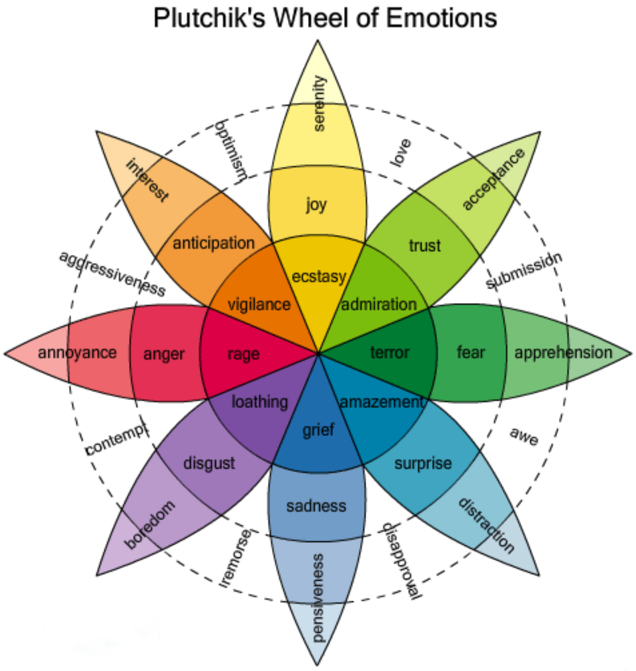


FIGURE 2.1: Plutchik's Wheel of Emotions

is treated as a primary emotion in Parrot's emotion taxonomy (see Table 2.1). It is interesting to note that all the popular emotion theories commonly identify emotions such as *anger, fear, joy, sadness and surprise*. Further it helps in evaluating the performance of different computational models for emotion detection which predict the association between real world data (i.e. text) and the predefined emotion classes derived from the emotion theories.

2.1.4 Other Emotion Theories

Apart from the aforementioned emotion theories other less prominent ones proposed in psychology are summarized in Table 2.2 [27]. These emotion theories do not define any structures such as wheels (Plutchik) or taxonomies (Parrot) in order to organise and relate different emotions. Also emotions identified in these theories largely intersect with the emotions identified by the earlier theories. Therefore in computational studies of emotion [28–30] Ekman, Plutchik and Parrot emotion theories are more commonly adopted given their wide expressiveness in different domains such as Twitter, news feeds, incident reports etc.

Theorist	Basic Emotions
Arnold	Anger, aversion, courage, dejection, desire, despair, fear, hate, hope, love, sadness
Frijda	Desire, happiness, interest, surprise, wonder, sorrow
Gray	Rage and terror, anxiety, joy
Izard	Anger, contempt, disgust, distress, fear, guilt, interest, joy, shame, surprise
James	Fear, grief, love, rage
Oatley and Johnson-Laird	Anger, disgust, anxiety, happiness, sadness
Panksepp	Expectancy, fear, rage, panic
McDougall	Anger, disgust, elation, fear, subjection, tender-emotion, wonder
Tomkins	Anger, interest, contempt, disgust, distress, fear, joy, shame, surprise
Watson	Fear, love, rage
Weiner and Graham	Happiness, sadness

TABLE 2.2: Emotion Theories

2.2 Approaches for Emotion Modelling

Apart from defining models and taxonomies to enumerate and organise emotions, research in psychology also identified two major approaches for emotion modelling namely: *categorical* and *dimensional* [25]. We discuss the approaches for emotion modelling in the following sections.

2.2.1 Categorical Approach

In this approach emotions are modelled as distinct emotion classes. These emotion classes are induced from Ekman, Parrot and Plutchik emotion theories. Therefore emotion detection in a categorical approach is analogous to text classification into emotions using machine learning [21, 31]. However a categorical approach is not able to capture mixed emotions because it classifies each emotion-bearing expression into a single category. Also a categorical approach does not capture the intensity of an emotion-bearing expression, since the emphasis is on identification rather than quantification.

2.2.2 Dimensional Approach

In this approach emotion states are treated as being more related to one another as opposed to being independent. In other words an emotion is viewed as a point in a continuous multidimensional space where each aspect or characteristic of an emotion is represented as a dimension. Affect variability is captured by three dimensions namely valence, arousal and power [32]. Here valence (pleasure - displeasure) depicts the degree of positivity or negativity of an emotion. Whilst arousal (activation- deactivation) depicts the excitement or the strength of an emotion. The dimensional approach depicting emotions in the valence arousal 2D space is shown in Figure 2.2 [1]. For example *joy* exhibits higher levels of excitement and positivity compared to *surprise*. Similarly *sadness* is more apathetic and negative compared to *anger*.

A dimensional approach follows exactly the principles of a lexicon based approach for emotion or sentiment classification wherein each piece of text is quantified with numerical values signifying the orientation of the text across different affective dimensions. Valence is captured by the membership of a given term to an emotion class and the lexicon score represents the arousal or strength of the emotion. The dimensional approach for emotion modelling is very relevant to the computational approach proposed in this research for emotion detection. The lexicon based methodology proposed for emotion detection in this research is used in conjunction with the emotion-sentiment mapping in Figure 2.2 to adopt an emotion corpus for sentiment analysis. We present the methodological details of this and the corresponding empirical evaluation in Chapters 6 and 7.

2.2.3 Emotion-Sentiment Relationship and its Application

The concepts of emotions and sentiments are used interchangeably due to the commonality in the biological and cognitive experiences they create. Research in psychology argues them as both related [1] and distinguishing experiences [33, 34]. However the theoretical frameworks established in psychology needs to be validated with real-world data driven experiments

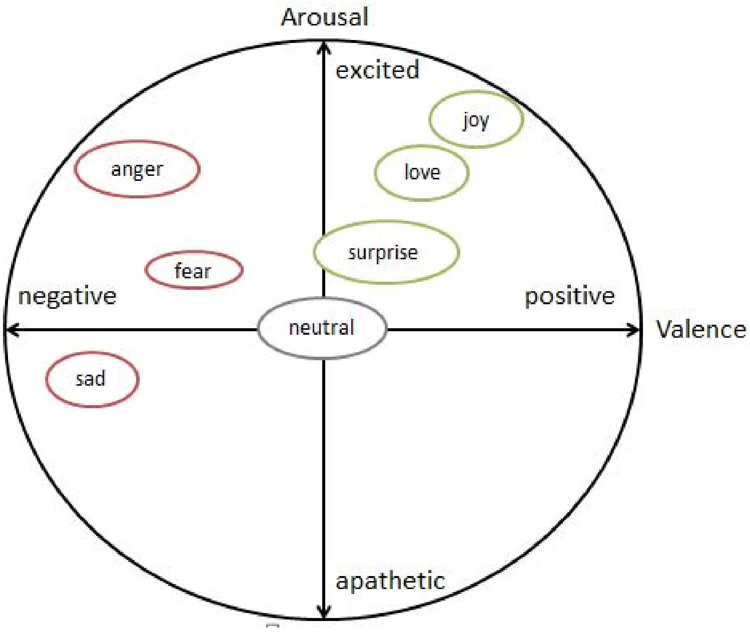


FIGURE 2.2: Emotions in the Valence-arousal plane of the dimensional model [1]

to conclude if the two concepts are independent or interdependent. Existing lexical resources for emotion detection [5, 14] and computational studies in emotion detection [28] suggest that emotion and sentiment bearing expressions are used together to convey views and opinions in real world text. However the role of emotion knowledge for sentiment analysis is limitedly explored [19, 35, 36] in computational studies for sentiment analysis. Therefore in this thesis we explore this direction by utilizing the theoretical mapping proposed between emotions and sentiments in psychology [1]. The mapping is used as a means to segregate emotion bearing expressions into positive and negative groups to augment the originally identified positive and negative expressions. In this thesis (refer Chapter 6) we validate the role of an emotion corpus coupled with a theoretical mapping between sentiment and emotions to learn computational models (emotion-aware sentiment lexicons) for sentiment analysis. Further we compare such models learnt with standard sentiment models that are ignorant to the knowledge provided by an emotion-rich corpus.

2.3 Computational approaches for Emotion Detection from Text

In the following sections we discuss the different approaches proposed for emotion detection from text. Broadly they can be classified as: *keyword-based, corpus-based and machine learning*. Further we also highlight the process followed in this thesis to adopt the emotion theories for computational study using lexicons and machine learning.

2.3.1 Keyword-based Methods

In this approach, text is modelled by observing the presence/absence of direct words that express emotions. There is usually a strong reliance on the availability of word sets that are manually organised into emotion categories. These manually generated lexicons can then be applied to compute the emotionality of a given piece of text on the basis of presence or absence of words in the lexicon. Text is classified into emotion categories based on the presence of emotion bearing words such as *distressed , enraged, and happy*. Elliott's Affective Reasoner [37], for example, uses a list of 198 emotion key-words (e.g. distressed, enraged), plus emotion intensity modifiers (e.g. extremely, somewhat, mildly), plus a handful of cue phrases (e.g. did that, wanted to).

Ortony's Affective Lexicon [38] provides an often-used source of emotion words grouped into affective categories. Using the affective lexicon alone for emotion detection has two weaknesses: poor recognition of emotion when negation is involved, and reliance on surface features. About its first weakness: while the approach will correctly classify the sentence, *today was a happy day*, as being happy, it is likely to fail on a sentence like *today wasn't a happy day at all*. About its second weakness: the approach relies on the presence of obvious emotion words which only amounts to surface level analysis of the text. In practice, a lot of sentences convey emotion through underlying meaning rather than emotion adjectives. For example, the text: *My husband just filed for divorce and he wants to take custody of my children away from me*, certainly evokes strong emotions, but uses no emotion keywords, and therefore, cannot be detected for emotion using a keyword spotting approach. However words such as *divorce*, *away* are emotion-bearing and modelling their occurrence patterns within emotion-labelled documents can potentially be more effective over keyword-spotting. In this thesis, we aim to model such word-level emotion associations in labelled emotion corpora using statistical language modelling in order to learn the emotions conveyed by sentences, paragraphs and documents.

2.3.2 Corpus-based Methods

Corpus-based methods for emotion detection apply supervised learning in order to induce knowledge sources such as word-emotion lexicons from a document corpus labelled or weakly-labelled with a predefined set of emotions derived from emotion theories such as Ekman, Parrot etc. Also unsupervised learning is adopted using external corpora such as a Wikipedia to model the syntactic and semantic patterns in text for emotion detection. However majority of works are lexicon-based, inspired by a significant amount of research in the related field of sentiment analysis. In the following sections we outline the different corpus-based methods for emotion detection from text.

2.3.2.1 Lexicon-based Emotion Detection

Similar to sentiment lexicons [12, 39–41], an emotion lexicon also contains a large (e.g. 40000) collection of words. However an emotion lexicon, unlike sentiment lexicons [12, 17, 42] offers granular information about the emotion orientation of words. Typically emotion lexicons capture the word-emotion associations either in the form of discrete labels or in the form of numerical scores. In this thesis, we refer to an emotion lexicon that captures word-emotion associations in the form of discrete labels as a general purpose emotion lexicon (GPEL). Further lexicons that capture word-emotion associations in the form of numerical scores are referred to as domain-specific emotion lexicons (DSELs). More formally a GPEL, $Lex(w, e_j)$ is a list of words for emotion class e_j as follows:

$$Lex(w, e_j) = \begin{cases} 1 & \text{if } w \in List(e_j), \\ 0 & \text{otherwise} \end{cases} \tag{2.1}$$

where $List(e_j)$ denotes the list of words corresponding to the j^{th} emotion from a pre-defined set of emotions E in the GPEL. In contrast to GPELs, a DSEL quantifies the associations between words in a vocabulary V and a set of pre-defined emotions E. More formally a DSEL

$Lex(w, e_j)$ is a numerical value which quantifies the association between the word w in vocabulary V and the emotion e_j as follows:

$$Lex(w, e_j) = \begin{cases} non-zero & \text{if } w \text{ occurs in documents labelled with emotion } e_j \\ 0 & \text{otherwise} \end{cases} \tag{2.2}$$

The exact value of $Lex(w, e_j)$ is determined by the lexicon generation adapted to learn a lexicon. For example probabilistic techniques like latent Dirichlet allocation assign a score in the range of [0,1]. For any given arbitrary word w, the dominant emotion e expressed is calculated using the lexicon as follows:

$$e = \arg\max_j Lex(w, e_j) \tag{2.3}$$

A GPEL is a static lexicon and needs human efforts in its creation, maintenance and modification. On the other hand, a DSEL is automatically generated from a document collection. Further it is also possible to efficiently model the variations in the vocabulary statistics of a DSEL in dynamic and evolving streams of data and update the word-emotion distributions accordingly.

Research in emotion detection resulted in development of both GPELs and DSELs. WordNet synsets [14] are manually labelled with Ekman basic emotions [22] to generate WordNet-Affect [4] (details are explained in Chapter 3). The NRC word-emotion lexicon [5] is obtained by crowd sourcing (using Mechanical Turk) Plutchik emotion [24] annotations for 10000 words obtained from Google n-gram corpus[1] and General Inquirer [10]. Machine Learning techniques have been applied (refer Chapter 3) to assign WordNet-Affect emotion labels to concepts in SenticNet [41] to obtain EmoSenticNet [6]. A common limitation for the aforementioned emotion lexicons is that their vocabulary is static and formal, thereby making them less applicable in dynamic and informal domains (e.g. social media) for emotion detection. A DSEL [43, 44] on the other hand has the ability to model the domain closely in order to capture the emotional context of words. This is possible with the help of labelled emotion corpora by applying the principles of supervised and semi-supervised learning.

Existing methods for learning DSELs are mostly supervised, since they rely either on labelled or weakly-labelled emotive content in a domain. Weakly-labelled emotive content is adopted

[1]https://catalog.ldc.upenn.edu/LDC2006T13

for lexicon generation, following the significant contributions of weakly-labelled sentiment corpora to performance in sentiment analysis [18] and [45]. For instance [46] and [47] proposed similar approaches to learn a word-emotion lexicon from crowd-annotated emotional news articles[2,3] by combining the document-frequency distributions of words and the emotion distributions over documents. However, since human annotations are expensive to obtain, lexicon generation was targeted on weakly-labelled emotive content which is abundant in social media. In particular, Point-wise Mutual Information (PMI) was applied to capture the association between words and emotion-rich constructs, such as emotion hashtags [48] and emoticons (in Chinese weblogs) [49]. Whilst the approach proposed by [48] is applicable to any emotion labelled corpora, the approach by [49] is specifically suited to emoticon rich corpora.

In contrast to the above works which build discriminative emotion models, generative models like Latent Dirichlet Allocation (LDA) are also applied to lexicon generation. [50] proposed a semi-supervised LDA approach, which uses a minimal set of domain-independent emotion seed words to learn emotion-relevant topics, under the assumption that documents exhibit multiple topic (emotion) characteristics and words contained in documents also reflect the underlying topics. However the topics learnt from this approach are not consistently accurate, since the coverage of seed words varies from one domain to another. Nevertheless, supervised LDA (sLDA) [51] offers a more accurate means to learn emotion-topic models for lexicon generation, from labelled or weakly-labelled emotion corpora. We show later (see chapter 3) how emotion-topics from sLDA can be transformed into a lexicon. In this thesis we assume documents to be a mixture of emotional and neutral words, which is different from the generative model of sLDA that assumes documents to be a mixture of multiple emotion (topic) words. A similar mixture model with two components was found to be effective in characterising problem-solution documents [52, 53]. We expect the joint modelling of emotionality and neutrality at word-level to be more effective on real-world emotion corpora, since not every word in them connotes emotions. Further, since emotion corpora in general have short or medium sized documents, we expect the proposed mixture model to characterise them better than sLDA.

[2]http://news.sina.com.cn/society/
[3]http://www.rappler.com/

2.3.2.2 Knowledge-augmented Corpus-based Methods

Knowledge-augmented corpus-based methods for emotion detection apply unsupervised learning, using external corpora such as Wikipedia [4], Gutenberg [5] and British National Corpus (BNC) [6] to learn semantic relationships between general words and emotion bearing words. For instance, [54] proposed to learn semantically related words to those present in WordNet-Affect using Latent Semantic Analysis (LSA). Each WordNet-Affect synset is represented as vector in the LSA space learnt from the BNC corpus. This vector space is used to find semantic similarities between words, with an intent to expand the word lists in WordNet-Affect. The expanded word-list resulted in performance improvements over using WordNet-Affect alone in emotion classification of blog sentences and news headlines.

Similarly [55] proposed a novel unsupervised context based approach to detect emotions at sentence level in blogs, fairy tales [56] and incident narratives [57]. The proposed method identifies NAVA (nouns, adjectives, verbs and adverbs) in the text (sentence) and measures the semantic relatedness of each with respect to an emotion using a set of seed words (e.g *happiness : glad, joy, good; anger: irritate, stupid, frustrate*) that correspond to the emotion. Large text corpora such as Wikipedia [4] and Gutenberg[5] were used to measure semantic similarity between words (NAVA, emotion seed words) using PMI in order to learn the values for the emotion vectors corresponding to the NAVA words. Further a set of rules were also proposed to account for contextual emotion analysis. The proposed methods resulted in significant performance improvements over simple keyword-spotting approaches and WordNet-Affect alone. However both the above mentioned approaches rely on generic text corpora to expand their emotion word lists, making them agnostic to the domain-level subtleties in emotion elicitation. Further the inability to quantify word-emotion associations also limits their usefulness in tasks such as word-emotion ranking and feature extraction for emotion text classification using machine learning.

[4] http://download.wikimedia.org/enwiki/latest/enwiki-latest-pages-articles.xml.bz2
[5] http://www.gutenberg.org
[6] http://www.natcorp.ox.ac.uk/

2.3.3 Machine Learning for Emotion Detection from Text

A majority of the literature concerning emotion detection, emotion classification in particular is shaped by machine learning approaches. These approaches represent documents as vectors in a feature space and classify them into predefined emotion categories defined by emotion theories such as Ekman and Plutchik. The feature extraction process for emotion classification is summarized in Figure 2.3. Observe that the lexicon based features are extracted using the knowledge of the DSEL learnt on the training documents. POS taggers, sentiment lexicons and GPELs act as external resources for extracting relevant features for emotion classification. Also Tables 2.3 and 2.4 summarize the state-of-the-art for emotion text classification. In the rest of the section we review in detail the state-of-the-art features proposed for emotion classification of text by organising them into different categories: *generic n-gram features*, *special n-gram features*, *lexicon-based features* and *additional features*.

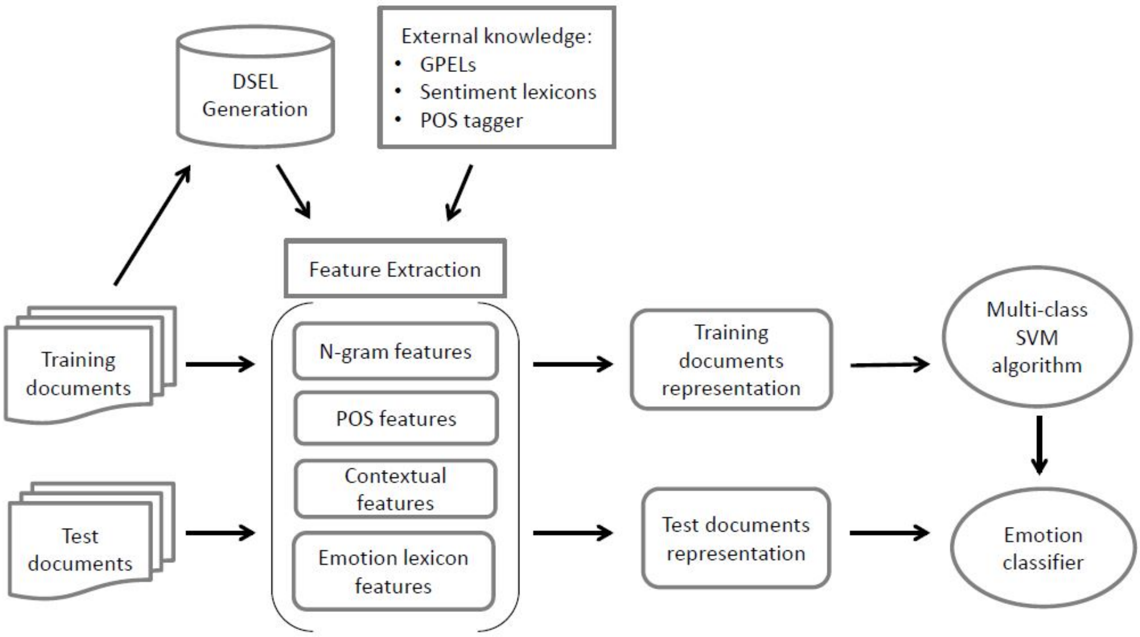


FIGURE 2.3: Feature extraction and emotion classifier learning

Generic n-gram features: This is the most standard representation used in text classification tasks including emotion classification. Documents are represented in a space of unordered list of terms (BoW or n-grams) as vectors. [58] used n-grams as features with tf-idf[7] weights as

[7] http://en.wikipedia.org/wiki/Tf-idf

feature values to classify Czech news headlines. Similar to the findings in sentiment classification [59], [60] and [61] demonstrated the effectiveness of n-gram features with binary weighting (word presence/absence) in emotion classification of blogs and tweets respectively. However a common limitation of n-gram features is their inability to capture the underlying emotion semantics, thereby resulting in overall performance degradation. This has lead to research [62, 63] which explores richer features that are better suited for emotion classification.

Study	Features	Data	Classes
[58]	BoW (TF-IDF)	Czesh news headlines	Ekman and neutral
[62]	Emotion concept features from WordNet-Affect and Manual emotion word lists	news headlines	Ekman
[60]	BoW(binary), presence/absence of emotion words in WordNet-Affect and Roget's thesaurus	blog sentences	Ekman and neutral
[64], [48]	BoW(binary), No of words associated with an emotion using NRC emotion lexicon and the PMI-lexicon (Section 2.3.2)	news headlines, blogs	Ekman
[63]	BoW(binary), presence/absence of emotion words in WordNet-Affect, MPQA subjectivity lexicon [40], Roget's thesaurus [65]	blogs	Ekman

TABLE 2.3: Emotion classification of blogs and news headlines

Study	Features	Classes
[66]	corpus n-grams(n = 1,2,3), topic scores from LDA, highly similar unigrams w.r.t emotion seed words, presence/absence of !,?	Ekman emotions and love
[61]	corpus n-grams (n =1)	Plutchik's emotions
[28]	corpus n-grams (n = 1,2,3), positional n-grams, % of POS words , presence/absence of emotion words in LIWC, MPQA subjectivity lexicon and WordNet-Affect	Parrot's primary and thankfulness
[67]	corpus n-grams(n=1), presence/absence of emotion words using emotion hashtag lists	Parrot's primary emotions
[68]	corpus n-grams(n=1)	Ekman emotions

TABLE 2.4: Emotion Classification of Tweets

Special n-gram features: As alluded to earlier, specialized features (e.g. punctuation) have been explored in the case of emotion analysis, as in the case of other specialized tasks such as author identification. These features were designed to capture the emotive expressions that occur in subtle ways, especially in Twitter. For instance, [28] designed features such as positional n-grams (i.e. n-grams in the first half of a tweet and n-grams in the second half of a tweet) and part-of-speech (POS) tagging to complement generic n-grams for emotion classification of tweets. Similar to the findings in sentiment classification [59] positional n-grams decreased performance, whilst POS information lead to marginal improvements over n-grams in emotion classification. [66] observed that modelling the presence/absence of punctuation (!, ?) marginally improves classification performance for emotions such as *surprise* and *joy* on tweets.

Lexicon based features: These features are designed based on the intuition that sentiment/emotion bearing words identified by lexicons can form useful knowledge to represent documents for emotion classification. [60] augmented generic n-grams with features to count the occurrences of emotion words provided by GPELs to significantly improve emotion classification of blogs. Whilst GPELs offer useful knowledge about emotion-rich words, they are static and are likely to have poor coverage of the emotion vocabulary used in domains like Twitter. For emotion classification of tweets [5] and [48] demonstrated that DSEL based features offer significant gains over n-grams when compared to those of GPEL based features [28]. However feature extraction using DSELs has not been explored beyond binary and integer counts. In particular the knowledge of a DSEL to quantify the association between words and emotions can be leveraged to design more sophisticated features for emotion classification. In this thesis we aim to explore the knowledge of a DSEL, to propose different feature extraction techniques, that can potentially improve performance in emotion text classification.

Additional features: Apart from the aforementioned features additional knowledge sources such as emotion hashtags [67], emotion word lists [62], topic scores [66] were used to design features that complement the n-gram features and general purpose lexicons such as WordNet-Affect. Performance improvements were observed in emotion classification tasks over using n-grams alone [67], but were found to be less effective when compared with lexicon based features suggesting that lexicon based features need to be explored further to design better and more effective text representations for emotion classification of text. In this thesis we further explore the potential of DSELs to extract effective representations for emotion classification. We also evaluate the contributions of the proposed features by comparing their performance with existing state-of-the-art features in emotion classification.

2.3.4 Adapting Emotion Theories for Computational Study

In order to study the problem of computational emotion detection the fundamental requirements are:

1. Textual data that expresses human emotions identified by the different emotion theories discussed earlier
2. Computational models capable of automatically detecting the emotions expressed

Existing works in computational emotion detection already demonstrate the possibility of developing models to automatically detecting emotions in text [60], [5] and [48]. However none of the existing works highlight how the emotion theories proposed in Psychology were adopted for their computational study. In this section we attempt to establish the process used to choose an emotion theory for computational study of detecting emotions from text. Firstly we reviewed the research work in computational emotion detection for benchmark data sets that are publicly available for research. After gathering the data sets we reviewed the emotions represented across the data sets and also their corresponding emotion theory links. We observed that most of the non-social media data sets that are publicly available (refer section 3.6 in Chapter 3) represent four emotion theories: Ekman, Parrot, Izard and Plutchik. Further in the case of the Twitter data set [28] which was re-crawled using the Twitter API services we found that the primary emotions identified by Parrot emotion theory are well represented to form a sizeable sample for developing computational models. Therefore the choice of emotion theories for computational study in this research is primarily influenced by data representation they have in real world data sets available for research.

2.4 Emotion Knowledge for Sentiment Analysis

Sentiment analysis concerns the computational study of natural language text (e.g. words, sentences and documents) in order to identify and effectively quantify its polarity (i.e positive or negative) [7]. Sentiment lexicons are the most popular resources used for sentiment analysis, since they capture the polarity of a large collection of words. These lexicons are either handcrafted (e.g. opinion lexicon [9], General Inquirer [10] and MPQA subjectivity lexicon [11]) or generated (e.g. SentiWordNet [12] and SenticNet [13]) using linguistic resources such as WordNet [14] and ConceptNet [15]. However, on social media (e.g. Twitter), text typically contains special symbols resulting in non-standard spellings, punctuations, capitalization, sequence of repeating characters and emoticons for which the aforementioned lexicons have limited or no coverage.

As a result domain-specific sentiment lexicons were developed to capture the informal and creative expressions used on social media to convey sentiment [16, 17]. The extraction of such lexicons is possible with limited effort, due to the abundance of weakly-labelled sentiment data on social media, obtained using emoticons [18, 19]. However, sentiment on social media is not limited to conveying positivity and negativity. Socio-linguistics suggest that on social media, people express a wide range of emotions such as *anger, fear, joy, sadness* etc [69]. Following the trends in lexicon based sentiment analysis, research in the textual emotion detection also developed lexicons that can not only capture the emotional orientation of words [5, 70], but also quantify their emotional intensity [43, 46].

Though research in psychology defines sentiment and emotion differently [34], it also provides a relationship between them [31]. Further research in emotion classification [28, 63] demonstrated the usefulness of sentiment features extracted using a lexicon for document representation. Similarly emoticons used as features to represent documents improved sentiment classification [16, 19]. However, the exploration of emotion knowledge for sentiment analysis is limited to emoticons [19, 35, 36], leaving a host of creative expressions such as emotional hashtags (e.g. #loveisbliss), elongated words (e.g. haaaappyy!!!) and their concatenated variants unexplored. An emotion-corpus crawled on Twitter using seed words for different emotions as in [28, 48] can potentially serve as a knowledge resource for sentiment analysis. Adopting such corpora for sentiment analysis, e.g. sentiment lexicon extraction is particularly interesting, given the challenges involved in developing effective models which can cope with the lexical variations on social media.

Therefore it is interesting to study the role of such emotion knowledge for sentiment analysis, in particular for sentiment lexicon generation and validate its usefulness. Here we focus on this aspect, by exploiting an emotion-labelled corpus of tweets to learn sentiment lexicons. We achieve this by combining our prior work on generative mixture models for lexicon extraction and the emotion-sentiment mapping provided in psychology (refer figure 2.2).

2.5 Conclusions from the Literature

It is very evident from the literature that lexicon based and machine learning based methods are widely used for emotion detection in text. In particular in machine learning approaches a combination of corpus level features and features derived from lexicons such as WordNet, WordNet-Affect and NRC lexicon are used to extract relevant text representations for emotion classification. Though WordNet-Affect, NRC lexicon and EmoSenticNet are developed to aid emotion detection, the formal and static nature of these make them ineffective in domains (social media, blogs) where the vocabulary is constantly evolving. One way to overcome this challenge is to develop domain specific emotion lexicons that effectively capture the emotive expressions. Though there exists research on domain-specific emotion lexicon generation, the proposed methods do not effectively capture the characteristics of real-world emotion data. Methods proposed in [46] and [47] are explicitly designed for document corpus with emotion ratings. Further methods proposed using PMI [48] and LDA [50] suffer from inabilities to model low-frequent emotion relevant words and word-mixtures in emotional short text (e.g. tweets, news headlines, blog sentences) respectively. In this thesis, we address this problem of learning a domain-specific word-emotion lexicon by proposing a novel word (unigram) mixture model, whose parameters are optimized using Expectation Maximization (Chapter 4).

It is also observed in the literature, the knowledge of a word-emotion lexicon is limitedly utilized to extract features to represent documents for emotion classification (eg. emotion word counts in the text using a lexicon). However counting based features are not sufficiently effective in detecting all emotions, thereby impacting overall performance. In this thesis, we seek to use the numerical scores offered by the DSELs to derive additional features, which we expect to be more effective for discriminating emotions. Further we would evaluate the quality of such derived features by comparing their performance in classification tasks with state-of-the-art text representations used in emotion classification (Chapter 7).

Research in sentiment analysis found improvements in performance when emotion-related knowledge is utilized in the learning of the sentiment model [16, 19]. However, in sentiment analysis of social media, the emotion knowledge explored is only limited to emoticons [19, 35, 36]. Therefore in this thesis (Chapter 6), we explore the role of an emotion-corpus crawled on Twitter using seed words for different emotions as in [28, 48] as a potential knowledge resource for sentiment analysis. Adopting such corpora for sentiment analysis, e.g. sentiment lexicon extraction is particularly interesting, given the challenges involved in developing effective models which can cope with the lexical variations on social media.

2.6 Chapter Summary

In this chapter we reviewed the literature in emotion detection that is relevant to our work. We discussed the various emotion theories proposed in Psychology and also highlighted those theories which identify the most widely expressed emotions in real-world emotional text. Thereafter we reviewed the different computational approaches for emotion detection: key-word based, corpus-based and machine learning. The review focussed on research progress in each of the approaches, their strengths and weaknesses. Finally we reviewed the research done in exploring the role of various forms of emotion knowledge in the field of sentiment analysis, highlighting the research gap, which this thesis aims to address.

Chapter 3

Background

In this chapter we begin by presenting the details of different general purpose emotion lexicons (GPELs) and domain specific emotion lexicons (DSELs) which form the baselines in our comparative study. Thereafter we present details of the baseline feature extraction methods used to represent text for emotion classification. Finally we provide details about the datasets and performance metrics used in our evaluation.

3.1 General Purpose Emotion Lexicons

Word-emotion lexicons are powerful resources for emotion detection from text, since they associate the relationships between words and different emotions. These relationships further can be used to extract effective features to represent documents for emotion classification. In literature the first kind of lexicons developed for emotion detection from text, were either manually crafted (e.g. WordNet-Affect, NRC emotion lexicon) or obtained using semi-automatic methods (e.g. Emosenticnet). The common characteristic for all these lexicons is that their vocabulary is static (i.e domain agnostic) and also they account significantly for formal and standard words in English. In this thesis we refer to these lexicons are GPELs and use them in a comparative performance evaluation along with other DSELs.

3.1.1 WordNet-Affect

WordNet-Affect is derived from a general knowledge database known as WordNet [14]. The synsets in WordNet are a set of terms that share similar meaning. All synsets that represent emotion concepts are selected and labelled with emotions. Further each word in WordNet is also tagged with POS information. Therefore a synset in WordNet-Affect is the set of words that are synonymous and that convey the same emotion.

WordNet-Affect lexicon is a list of terms tagged with POS information representing the Ekman emotions. WordNet-Affect contains 2874 synsets and 4787 words. A sample list of terms in the WordNet-Affect lexicon is shown in Table 3.1. Given the emotion in the first column, a synset index to synonyms appears in the second column, followed by the synset in the last column. The POS tags *n, v, a, r* denote noun, verb, adjective and adverb respectively.

3.1.1.1 Development of WordNet-Affect

The two-staged development of WordNet-Affect involves the initial resource gathering, followed by its expansion.

Emotion	SynsetID	Words
Anger	n#05614716	irascibility, short-temper, spleen
Fear	v#01214618	frighten, fright, scare, affright
Joy	a#01215015	hilarious, screaming, uproarious
Sadness	r#00359722	penitently, penitentially, repentantly
Surprise	a#01230203	astonishing, astounding, staggering
Disgust	r#00304232	detestably, repulsively, abominably

TABLE 3.1: Sample terms in WordNet-Affect Lexicon

1. *Core WordNet-Affect Creation*: A manually created lexical data base called AFFECT is used for this purpose. It contains 1903 words which refer to mental (e.g. emotional) states directly or indirectly. AFFECT predominantly contains nouns (539), adjectives (517), followed by verbs (238) and adverbs (15). For each word in AFFECT lexical and affective information is added. Lexical information includes the correlation between English and

Italian terms, POS , definitions, synonyms and antonyms. Similar to synsets the POSR attribute identifies words having different POS but pointing to the same psychological category. Affective context of a word captures its membership into different categories proposed in [71]: *emotion, cognitive state, trait, behaviour, attitude and feeling*. Once the lexical and emotion context is established, each word's corresponding synset from WordNet is gathered to form the affective core. This essentially means each synset identified represents an affective concept. Further processing ensures filtering of synonyms with incompatible values of the affective information in each synset. The development process for WordNet-Affect is visualized in Figure 3.1.

2. *Extension of the Core WordNet-Affect*: WordNet offers many different lexical and semantic relations between words and synsets. These relations are used to propagate the affective information from the core synstes to other synsets. After a manual check for preservation of affective information relations such as antonymy, similarity, derived-from, pertains-to, attribute, also-see are identified as reliable, whereas relations such as hyponymy, entailment, verb-group etc are identified as partially reliable relations. After propagation and filtering the final resource WordNet-Affect with 2874 synsets and 4787 words is obtained.

Though WordNet-Affect resulted in the expansion of the relatively small AFFECT lexicon, it still has limited coverage for the emotion vocabulary used in real world data. Therefore additional resources were developed with an objective to have improved coverage lexicons for emotion detection from text. In the following sections, we present the details of two such emotion lexicons which are generated using the vocabulary of WordNet-Affect.

3.1.2 NRC Word-Emotion Association Lexicon

NRC word-emotion lexicon is developed using the principle of crowd sourcing. In order to generate the 10,000 sized word-emotion lexicon, initially a list of words and phrases are identified using resources such as Macquarie thesaurus, WordNet-Affect lexicon and General Inquirer [10]. These resources are chosen, since they cover a wide list of emotion-related words. Words from Macquirie thesaurus are further reduced by selecting only those which overlap with the Google n-gram corpus. Finally a set of words/phrases is created by taking the union of the

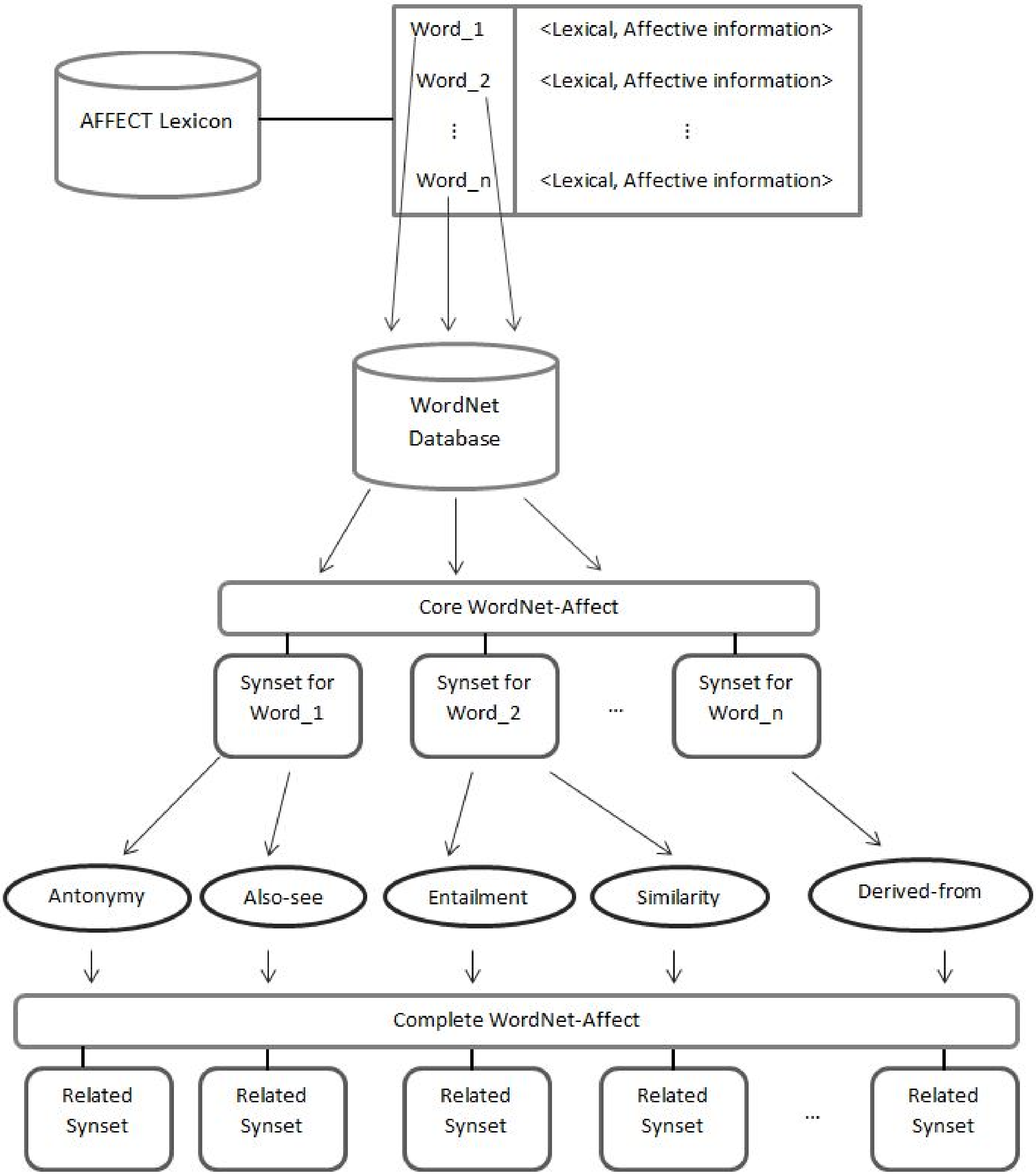


FIGURE 3.1: Illustration of WordNet-Affect generation

NRC lexicon	# terms	% of the union
	Unigrams from Macquarie thesaurus	
adjectives	200	2.0%
adverbs	200	2.0%
nouns	200	2.0%
verbs	200	2.0%
	Bigrams from Macquarie thesaurus	
adjectives	200	2.0%
adverbs	187	1.8%
nouns	200	2.0%
verbs	200	2.0%
	Terms from General Inquirer	
negative terms	2119	20.8%
neutral terms	4226	41.6%
positive terms	1787	17.6%
	Terms from WordNet-Affect Lexicon	
anger terms	165	1.6%
disgust terms	37	0.4%
fear terms	100	1.0%
joy terms	165	1.6%
sadness terms	120	1.2%
surprise terms	53	0.5%
	Total terms in NRC Lexicon	
Union	10170	100%

TABLE 3.2: Target term statistics in the NRC Lexicon

aforementioned resources. Only those words having at most three senses are selected. Statistics about the contribution of each resource towards the word list is shown in Table 3.2.

In the second stage, thousands of volunteers are presented with a questionnaire using Amazon's mechanical turk. The first question tests a participant's English proficiency involving synonym selection and is aimed to establish annotator confidence. The follow on questions are related to measuring the association of the word presented in the first question with [24] emotions and sentiments. A sample of the questionnaire is shown below:

1. Q1: What is the closest word in meaning to *startle*?

    - *auto-mobile*
    - *shake*
    - *honesty*

Word	Anger	Anticipation	Disgust	Fear	Joy	Sadness	Surprise	Trust	Negativity	Positivity
abnormal	0	0	1	0	0	0	0	0	1	0
provoking	1	0	1	0	0	0	0	0	1	0
reassure	0	0	0	0	0	0	0	1	0	1
punch	1	0	0	1	0	1	1	0	1	0
muck	0	0	1	0	0	0	0	0	1	0
revolution	1	1	0	1	0	1	1	0	1	1
unclean	0	0	1	0	0	0	0	0	1	0

TABLE 3.3: Sample terms in NRC word-emotion Lexicon

- *entertain*

2. Q2: How *positive* is the word *startle* ?

    - *not positive*
    - *weakly positive*
    - *moderately positive*
    - *strong positive*

3. Q2: How much is the word *startle* associated with emotion *joy* ?

    - *not associated*
    - *weakly associated*
    - *moderately associated*
    - *strongly associated*

Responses from this questionnaire are used to generate the NRC lexicon (see an example in Table 3.3). Here each term is given a score of 1 if it represents a particular Plutchik's emotion/sentiment and 0 otherwise. NRC lexicon is an authentic resource for emotion detection, as it is obtained from the knowledge of several humans, with over 10,000 words. It also has wider coverage than lexicons such as WordNet-Affect.

3.1.3 EmoSenticNet

EmoSenticNet is a lexical resource obtained by extending the WordNet-Affect emotion labels to SenticNet concepts automatically. Though WordNet-Affect and NRC emotion lexicon capture word-level emotion associations, they have limited coverage for concepts (i.e. human common-sense knowledge), which are also used in text to express emotion. In sentiment analysis concept-level lexicons such as SenticNet [41] developed using resources such as ConceptNet [15], resulted in performance improvements. Similarly research in emotion detection also focussed on developing concept-level emotion lexicon (i.e. EmoSenticNet). Beginning with a seed list of concepts with emotion labels present in both SenticNet and WordNet-Affect, the emotion labels are learnt for the remaining concepts in SenticNet using a combination of unsupervised and supervised algorithms.

3.1.3.1 Generation of EmoSenticNet

The generation of EmoSenticNet is done in six steps as follows:

1. *Identification of Seed Emotion Concepts*: An initial set of concepts are identified using lexical resources such as SenticNet and WordNet-Affect. SenticNet is a concept-level lexicon which assigns sentiment scores for single and multi-word concepts derived from ConceptNet (a lexical resource representing common-sense knowledge). The objective in this step is to identify SenticNet concepts that are also present in WordNet-Affect, in order to form a repertoire of emotion labelled concepts. A total of 1128 SenticNet concepts are present in WordNet-Affect. These concepts are used to train a supervised SVM classifier which assigns emotion labels to the remaining concepts in SenticNet.

2. *Feature Vector Extraction for Concepts*: In this step each concept is represented as a feature vector, in order to apply clustering and classification algorithms for learning emotion labels. Broadly the features used to represent concepts are: *ISEAR data-based features, similarity measure-based features*. ISEAR data is used to extract physiological, behavioural and emotional information corresponding to concepts. A total of 16 different features are extracted for each concept from the ISEAR dataset. Similarity based features

are used to identify how related a concept is to the remaining concepts. Metrics such as Word-Net distance, SenticNet distance, ISEAR text distance are used to compute similarity scores between all possible concept pairs. Additionally point-wise mutual information (PMI) is also used as a metric to extract similarity scores for concept pairs. Finally each concept is represented as a feature vector, representing information obtained from ISEAR data and the different similarity measures.

3. *Clustering of Concept Feature Vectors*: Fuzzy k-means clustering is applied on the feature vectors to cluster them into six groups. Instead of a hard assignment of a concept to a single cluster, each concept vector is assigned a membership score for each of the six clusters. This is done to preliminarily estimate the emotion class of each concept, which is refined in the steps 5 and 6.

4. *Mapping Fuzzy Classes to Emotion Labels*: First a hard cluster for each concept vector is identified based on the strength of the membership scores. Thereafter the seed emotion concepts present in each hard cluster are used on a majority vote basis to assign an emotion label for each cluster. The assignment of an emotion label for each concept is done using supervised classification with SVM, in step 6.

5. *Cluster Membership Restriction and Feature Vector Extension*: In this step, for each concept, the top K (K is empirically estimated) clusters are identified based on its membership scores. Thereafter the feature vector for each concept is appended with the top K membership scores to obtain a new feature vector. This essentially adds the emotion class knowledge into the feature vector which will be utilized by the SVM classifier in the final emotion class prediction for concepts.

6. *Final Hard Classification*: In this step, SVM classifiers are trained separately for each of the $\binom{6}{K}$ possible combinations of the K emotion labels. The training is done using just the seed concepts from WordNet-Affect for which there are assigned emotion labels. The trained classifier is then used to classify the feature vectors of the remaining concepts in SenticNet into [22] emotions, to finally obtain EmoSenticNet. A sample of EmoSenticNet is shown in Table 3.4. Observe that the GPELs in general (see Tables 3.1, 3.3 and 3.4) capture standard and formal English words that convey emotion. However they do not account for the domain-level context in which these words are used for emotion elicitation.

Word	Anger	Disgust	Fear	Joy	Sadness	Surprise	Sentiment-score
recreation	0	0	0	1	0	0	0.624
gift	0	0	0	1	0	0	0.909
disaffection	1	0	0	0	0	0	-0.400
agitation	1	0	0	0	0	0	-0.794
fell	0	1	0	0	0	0	-0.671
disorder	0	0	0	0	0	1	-0.532
detachment	0	0	0	0	1	0	-0.300
sinking	0	0	1	0	0	0	-0.123

TABLE 3.4: Sample terms in EmoSenticNet Lexicon

Words	Anger	Fear	Joy	Sadness	Surprise
PMI lexicon					
:)	-0.279	0.157	0.217	-0.241	-0.100
good!!	-0.182	0.254	-0.122	0.214	-0.003
#arrogant	0.419	0.458	-0.724	0.059	0.200
WED lexicon					
:)	0.157	0.064	0.515	0.136	0.121
good!!	0.193	0.055	0.417	0.159	0.172
#arrogant	0.187	0.065	0.464	0.128	0.156
sLDA lexicon					
:)	0.096	0.191	0.463	0.109	0.141
good!!	0.166	0.330	0.072	0.189	0.243
#arrogant	0.156	0.309	0.133	0.177	0.225

TABLE 3.5: A sample of the PMI, WED and sLDA word-emotion lexicons on Twitter emotion corpus

Further in domains such as social media, where the vocabulary is constantly evolving, the application of GPELs for emotion detection becomes extremely challenging. To alleviate such challenges several methods were proposed in the literature to learn DSELs.

3.2 Domain Specific Emotion Lexicons: Baseline Methods

3.2.1 Supervised Latent Dirichlet Allocation based Emotion Lexicon

Topic modelling algorithms aim to extract the important semantic structures (i.e. topics) in documents. The extracted knowledge is captured in the form of statistical models. Topic modelling is relevant for emotion detection, since an emotion can be modelled as a semantic concept which has certain characteristics and these change from one emotion to the other. Latent Dirichlet Allocation (LDA) [72] is a popular topic detection algorithm which models documents to exhibit characteristics of multiple topics. In sentiment analysis LDA is applied to capture the relationships between words and sentiment (positivity, negativity) in addition to the topics [73, 74]. Similarly in emotion detection, LDA has been applied in a semi-supervised manner using a minimal set of domain-independent seed emotion words to learn emotion-relevant topics [50]. However supervised LDA (sLDA) [51] offers a more accurate means to learn emotion-relevant topics from labelled/weakly-labelled emotion corpora, because the usage of a minimal set of seed emotion words, does not guarantee the same level of coverage for all domains, thereby affecting the accuracy of the topics generated.

Accordingly we can use sLDA to learn topic (emotion) distributions and map these into a word-emotion lexicon. More formally, let $\theta_{e_1}, \theta_{e_2}, \ldots, \theta_{e_n}$ be the topic distributions learnt for emotions $e_1, e_2, \ldots, e_n$, then the emotion lexicon is induced as follows:

$$sLDA_{Lex}(w_j, e_n) = \frac{P(w_j|\theta_{e_n})}{\sum_{i=1}^{|E|} P(w_j|\theta_{e_i})} \tag{3.1}$$

where θ_{e_n} is the topic distribution for emotion e_n obtained from sLDA, where w_j is the j^{th} word in the vocabulary V. Table 3.5 shows a sample of the sLDA based lexicon. Observe that for each word in the sLDA lexicon in Table 3.5 the word-emotion relationships are captured in the form of a probability distribution.

3.2.2 Point-wise Mutual Information based Emotion Lexicon

Point-wise mutual information (PMI) is generally used to quantify the discrepancy between the probability of coincidence of two events x, y, given their joint distribution and their individual distributions, assuming independence. In sentiment analysis PMI is commonly used to quantify the strength of association between a word and positive/negative sentiment, by modelling the occurrence patterns of words inside/outside documents that are labelled positive/negative[[75]]. Similarly for emotion detection, PMI has been applied to learn word-emotion lexicons, by modelling the occurrence patterns of words inside/outside documents that convey/not-convey the emotion [48]. In this research we use the PMI based word-emotion lexicon proposed in [48] as a comparative baseline. The generation method for the lexicon can be formally described as follows:

$$PMI_{Lex}(w_j, e_n) = Log\frac{freq(w_j, e_n) \times freq(\neg e_n)}{freq(e_n) \times freq(w_j, \neg e_n)} \tag{3.2}$$

where $freq(e_n)$ and $freq(\neg e_n)$ are the number of documents in X with and without emotion label e_n respectively. $freq(w_j, e_n)$ is the frequency of the j^{th} word in vocabulary V in documents labelled with emotion e_n and $freq(w_j, \neg e_n)$ is its counterpart. A sample of the PMI based lexicon is shown in Table 3.5. Observe that unlike the sLDA lexicon, which quantifies word-emotion relationships using probability values, the PMI lexicon quantifies word-emotion relationships with positive and negative values. High positive values indicate strong association, whereas negative values indicate disassociation.

3.2.3 Word-Document Frequency based Emotion Lexicon

Crowd-sourced emotion annotations provided by readers of the documents (e.g news stories) are used to learn word-emotion lexicon. These emotion annotations are in the form of numerical ratings, which can be normalized to define a probability distribution of emotions on each document. [46] proposed a lexicon generation method by combining the document-frequency distributions of words and the emotion distributions over documents. In this research we use the method proposed by [46] as a comparative baseline. Since this method involves modelling of frequencies of words in emotional documents and emotion ratings, we refer to the method as word-emotion-document (WED)lexicon. The generation method for the lexicon can be formally described as follows:

$$WED_{Lex}(w_j, e_n) = \frac{\sum_{i=1}^{|X|} P(w_j|d_i) r_{in}}{\sum_{n=1}^{|E|} \sum_{i=1}^{|X|} P(w_j|d_i) r_{in}} \tag{3.3}$$

where w_j is the j^{th} word in vocabulary V and r_{in} is the normalized emotion rating of the n^{th} emotion in E on d_i, the i^{th} document in the corpus X. A sample of the WED based lexicon is shown in Table 3.5. Observe that unlike the sLDA and PMI lexicons, the WED lexicon requires the emotion labels for the documents in the form of numerical ratings, thus making them suitable to apply on limited emotion corpora. However the common feature of all the DSELs in Table 3.5, unlike the GPELs is their ability to capture the associations between words used in a particular domain and different emotions. This gives the DSELs an ability to mine new expressions that are used to convey emotions in different domains.

3.3 Text Representation for Emotion Classification: Baseline Features

In this section we detail the commonly used features to improve emotion classification. Unlike the lexicon-based features these features do not rely on the knowledge of an emotion lexicon. We consider the following:

1. n-grams (n=1): These are the most standard corpus level features used in different classification tasks including sentiment [75] and emotion classification [60]. We used a binary weighting (presence/absence) to construct the feature vector, since it is found to be effective by earlier research in sentiment [59] and emotion classification [61].

2. Part-of-Speech (POS) features: Similar to [28], we used features to model the occurrence of verbs, adverbs, nouns and adjectives in a document. Part-of-speech tagging on non-social media data sets is done using the stanford POS tagger[1], whilst Twitter NLP tool [76] from Carnegie Mellon University is used for tagging social media data sets .

3. Contextual features (CF): Though standard words can convey the emotional intention of the author, additional expressions such as punctuation marks, emoticons are often used in social media to express emotions. Further sentiment bearing words could indicate the emotion in the text and also alter its orientation from positive-emotion(e.g. *joy*) to

[1]http://nlp.stanford.edu/software/tagger.shtml

negative-emotion (e.g. *sadness*) or vice versa. We consider the following popular contextual features used in sentiment [75] and emotion [28] classification for our comparative study:

- Capitalized words: This feature counts the number of words in a document with all upper case characters.
- Elongated words: This feature counts the number of words with character repeated two, three or four times. For example *haaappy*.
- Punctuation: Emotions are intensified on social media using exclamation marks and question marks. Two integer valued features are included to model the occurrence of question marks and exclamation marks in a document.
- Emoticons: Emoticons are facial expressions captured pictorially, and are often used on social media to convey emotions. A binary feature is designed to model the presence/absence of emoticons in a document. The emoticon list is adopted from an earlier work in emotion classification [77].
- Negation: Though the role of negation is not extensively studied for emotion classification, following its usefulness for sentiment classification [59], we include a feature to model the occurrence of negators in documents. We used a standard list of negators proposed by a popular work in sentiment analysis [78].
- Sentiment features: Though sentiment and emotion are different by definition [34], prior research in emotion classification [28] explored the role of sentiment knowledge offered by lexicons. Similarly we define two integer valued features, to capture the number of positive words and negative words observed in a document. However in addition to the sentiment lexicons used in [28] we consider more recent lexicons like SentiwordNet [12], SenticNet [13], NRC HashTag sentiment lexicon[2] and Sentiment140 lexicon[2], to have a wider coverage of sentiment bearing words across different domains.

The features discussed in this section are extracted to represent both training/test documents in each data set. The training feature vectors along with the emotion class labels are used to learn an SVM classifier in order to predict the emotion class label for unseen (test) documents.

[2] http://saifmohammad.com/WebPages/lexicons.html

3.4 Machine Learning Classifier : Support Vector Machines

A Support Vector Machine (SVM) belongs to the family of supervised machine learning algorithms. Intuitively an SVM tries to build a hyperplane in order to separate the given data points into two different classes. This is a classic example of binary text classification (e.g. sentiment classification of text). An SVM is also known as the maximum margin classifier, since it simultaneously tries to minimise the generalisation error while maximising the geometric margin

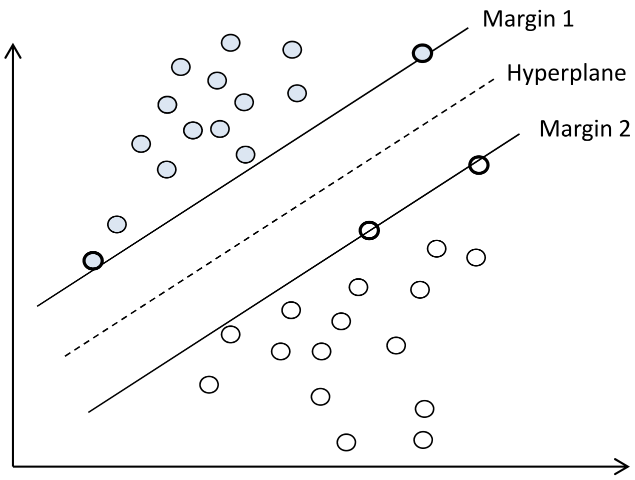


FIGURE 3.2: Support Vector Machines: Binary Classification

Here a separating hyperplane is constructed, followed by the maximization of the margin between the two classes. For calculating the margin, two parallel hyperplanes are constructed, one on each side of the initial one. These hyperplanes are then expanded perpendicularly away from each other until they are in contact with the closest training examples from either class. These examples are known as the support vectors and illustrated in bold in Figure 3.2. Intuitively, the best separation is the one with the largest margin between the two hyperplanes. Thus, the larger the margin; the lower the generalisation error.

In the case of multi-class classification, there are many ways in which it can be converted into several binary classification problems as proposed in [80]. One of the popular implementations of multi-class SVMs (LIBLINEAR) is found in [81], where a one-vs-rest approach is adopted, to convert the multi-class problem into K (number of classes) binary classification problems.

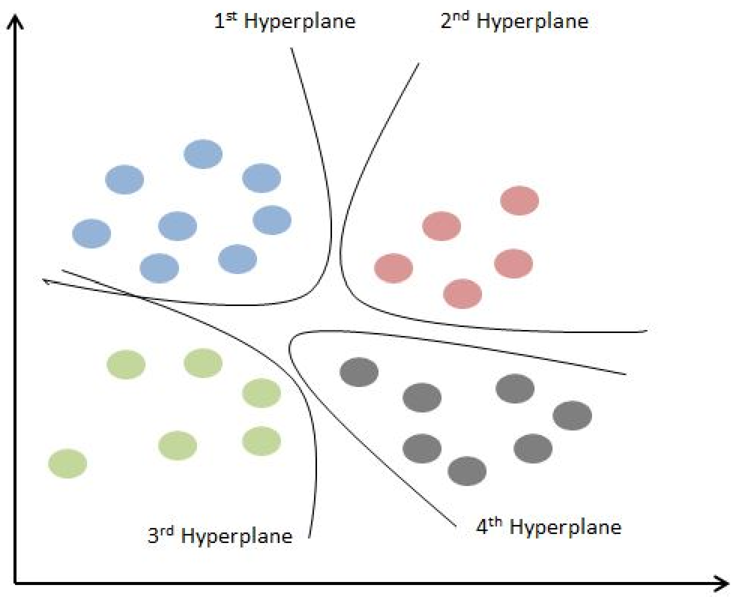


FIGURE 3.3: Support Vector Machines: Multi-class Classification

Here K hyperplanes are constructed to separate the data into K classes. An illustration of a 4-class SVM classifier is shown in Figure 3.3. Observe that the hyper planes shown in the figure are higher-dimensional, but projected into a two-dimensional plane for the purpose of visualization. SVM is a popular classifier used in text classification to achieve state-of-the-art performance [82],[83]. Therefore in this research we use SVM for emotion text classification.

3.5 Expectation Maximization for Text Mining

In this section we briefly explain the algorithm of expectation maximization (EM) and how it is applied in different text mining problems. EM is an iterative method to find the maximum likelihood or maximum a posterior estimates of parameters of statistical models [84, 85]. EM is typically applied for maximum likelihood estimation when the statistical model has parameters with hidden variables and a direct solution to the objective function of the statistical model cannot be obtained due to interdependencies between the parameters. In such cases EM is applied to iteratively estimate the values for the hidden variables based on a convergence criteria. EM is widely applied in the field of information retrieval to optimize the word-probability distributions that match relevant documents to user queries [86],[87]. Further EM is applied in the field of text segmentation to solve the parameters of models used in characterizing problem-solution documents [88].

3.6 Datasets and Statistics

In this section we describe the characteristics of the different data sets used in our evaluation. Some are useful for multiple emotion detection tasks (e.g. emotion ranking, emotion classification), whilst others are useful only for a particular task.

3.6.1 Emotion Detection Datasets

We present publicly available data sets that are either manually labelled emotion data sets or obtained using a distant supervision methodology, which is exploited to generate a data set automatically gather weakly-labelled emotion data. This is possible given the abundance of loosely tagged data (e.g. tweets) with emotion markers (e.g. emotional hashtags, emoticons etc) commonly on social media (e.g. Twitter). Also in the related field, Sentiment Analysis research has demonstrated the usefulness of distant labelled data to learn accurate supervised models [18],[45]. Therefore in this research we also leverage the availability of weakly-labelled emotion data on social media (e.g. Twitter) to learn word-emotion lexicons.

3.6.1.1 News data set (SemEval-2007)

A collection of 1250 emotional news headlines harnessed for evaluating the connection between emotions and lexical semantics at the SemEval-07 workshop [89]. Each headline was provided with emotion ratings in the range [-100, 100] for the Ekman basic emotions. We used this data set for emotion classification, by considering the highest rated emotion for each headline as the class label. Table 3.6 (columns 2 and 3) shows the distribution of different emotion classes in the training and test sets. The dataset is comparatively small with a considerable skewed class distribution. We are particularly interested to explore how the generative DSEL based features compare to baseline features. We expect that the smaller dataset size combined with the skewed distribution makes this an interesting dataset for comparison purposes.

Emotion	News (SemEval-07)		Twitter		Blogs		Incident Reports	
	# Training	# Test	# Training	# Test	# Training	# Test	# Training	# Test
Anger	67	23	57310	6496	140	36	816	204
Disgust	35	20	-	-	-	-	815	203
Fear	155	33	12592	1548	91	41	815	204
Joy	358	75	73098	8235	416	69	815	204
Sadness	201	61	62611	7069	136	57	815	204
Surprise	184	38	-	-	91	16	-	-
Love	-	-	30117	3464	-	-	-	-
Guilt	-	-	-	-	-	-	815	204
Shame	-	-	-	-	-	-	816	203

TABLE 3.6: Emotion Datasets

3.6.1.2 Twitter Dataset

A collection of 0.28 million emotional tweets[3] crawled from the Twitter search API using tweet identification numbers provided by [28]. Here emotion labels in the data set correspond to Parrot's primary emotions [23]. We used this data set for emotion classification (stratified 10-fold cross validation). Table 3.6 (columns 4 and 5) shows the average distribution of the different emotion classes over the 10 folds. As is evident from the table, not all emotions are strongly expressed in this data set. Emotions such as *joy, sadness* are more common compared to others like *fear, surprise*. Therefore it would be interesting to see how the different methods compare in performance given such class imbalance.

3.6.1.3 Blog Dataset

A collection of 5500 blog sentences annotated with Ekman basic emotions by 3 annotators with an average inter annotator agreement (kappa of 0.76) [30]. We used this data set for document classification using stratified 5 fold cross validation (not 10 fold due to the smaller size of the data set). Table 3.6 (columns 6 and 7) shows the average distribution of different emotion classes over the folds. The emotion class distribution is highly skewed towards the emotion *joy*. Further the smaller size of the data set is likely to challenge the modelling of the weakly represented emotions like *fear, surprise*.

[3] http://knoesis.org/?q=projects/emotion

3.6.1.4 Incident reports data set (ISEAR)

A collection of 7000 incident reports obtained from an international survey on emotion reactions [4]. Each report is an emotion summary, describing the situation which lead the participant to experience one of 7 emotions: *anger, disgust, fear, shame, guilt, joy and sadness*. We conducted a stratified 5-fold cross validation experiment on this data set. Table 3.6 (columns 8 and 9) shows the average distribution of different emotion classes over the 5 folds. Unlike the other data sets the emotion classes here have a near uniform distribution, which is very unlikely in a real word sample. It will also be interesting to observe how closely related emotions such as *shame* and *guilt* might be differentiated in the classification task.

3.6.1.5 Emotion event Dataset

A collection of 200 tweets describing emotional events [90] following Ekman basic emotions. Each event is annotated with a ranked list of emotions by two annotators with agreement (kappa of 0.68). We used this data set to test the quality of the lexicons on the emotion ranking task. Since this data set is very small, a lexicon learnt on the Twitter data was used here as both data sets are crawled from Twitter. We can also view this as a means to test the transferability of lexicons to different content albeit similar genre.

3.6.2 Sentiment Analysis Datasets

In this section we describe the different benchmark Twitter data sets that are available for experimental evaluation of sentiment analysis algorithms.

3.6.2.1 S140 Dataset

A collection of 1.6 million (0.8 million positive and 0.8 million negative) sentiment bearing tweets harnessed by [18] using the Twitter API. Further the data set also contains a collection of 359 (182 positive and 177 negative) manually annotated tweets.

[4] http://www.affective-sciences.org/researchmaterial

3.6.2.2 SemEval-2013 Dataset

A collection of 3430 (2587 positive and 843 negative) tweets hand-labelled for sentiment using Amazon Mechanical Turk [91]. Note that unlike the S140 test data, there is high skewness in the class distributions. Therefore it would be a greater challenge to transfer the lexicons learnt on the emotion corpus and also those learnt on the S140 training corpus to sentiment classification.

3.6.2.3 SemEval-2015 Dataset

A collection of 1315 words/phrases extracted from sentiment bearing tweets, hand-labelled for sentiment intensity scores [92]. A higher score indicates greater positivity. Further the words/phrases are arranged in decreasing order of positivity. We used this data set to validate the performance of different lexicons in ranking words/phrases for sentiment.

3.7 Evaluation Metrics

The evaluation metrics are chosen appropriately according to the emotion detection task. Further we also present the details of metrics used to estimate the optimal language models in our proposed method for lexicon generation.

3.7.1 Emotion Classification of Documents

In this research we comparatively assess the performance of different methods in emotion text classification, using metrics such as precision, recall and F-measure [93]. An illustration of a multi-class classifier outcome, given the human judgement in a five-class emotion (Anger, Fear, Joy, Sadness, Surprise) classification problem is shown in Table 3.7. Here the corresponding table of confusion for the class Anger is shown in Table 3.8. Where, TP is the number of angry documents correctly classified as angry (true positive), FP is the number of non-angry documents falsely classified as angry (false positive), TN is all remaining documents correctly classified as non-angry (true negative) and FN the number of angry documents falsely classified as non-angry (false negative).

Classification		Prediction				
		Anger	Fear	Joy	Sadness	Surprise
Actual Class	Anger	15	5	10	5	1
	Fear	4	33	14	6	0
	Joy	0	1	59	3	6
	Sadness	5	2	19	13	2
	Surprise	0	1	9	0	6

TABLE 3.7: Confusion Matrix

True Positive (TP):15	False Negative (FN):21
False Positive (FP):9	True Negative (TN):174

TABLE 3.8: Confusion table for Anger class

Precision for a given class $Anger$ is the fraction of correctly classified documents out of documents classified as $Anger$. Thus, the precision values for the class, $Anger$ (P_{Anger}) is determined as follows:

$$P_{Anger} = \frac{TP}{TP + FP} \tag{3.4}$$

Recall is the fraction of documents correctly classified out of all documents from a given class $Anger$. Therefore, recall for the class $Anger$, (R_{Anger}) is determined as follows:

$$R_{Anger} = \frac{TP}{TP + FN} \tag{3.5}$$

The F Measure for the class $Anger$ is obtained by taking the harmonic mean of the class' precision and recall as follows:

$$F_{Anger} = \frac{2 \times P_{Anger} \times R_{Anger}}{P_{Anger} + R_{Anger}} \tag{3.6}$$

We combine F measures from all classes, Anger (F_{Anger}), Fear (F_{Fear}), Joy (F_{Joy}), Sadness (F_{Sad}) and Surprise ($F_{Surprise}$), into a single value by taking their arithmetic mean as follows:

$$AvgF = \frac{F_{Anger} + F_{Fear} + F_{Joy} + F_{Sad} + F_{Surprise}}{5} \tag{3.7}$$

3.7.2 Emotion Ranking of Documents

In the task of document-emotion ranking, the quality of a method is assessed in terms of ordering the different emotions expressed by the document. Standard metrics such as Mean Reciprocal Rank (MRR) and Mean Average Precision (MAP) are used in this evaluation task. While MRR measures the quality of a method in predicting the dominant emotion present in the document, the ability of a method to order the multiple emotions expressed by a document is measured best by MAP. MRR is a standard metric used in the field of information retrieval to assess the quality of a retrieval algorithm in ordering the list of responses to a sample of queries. Similarly the quality of a method in predicting the dominant emotion expressed by a collection of documents D, is calculated as follows:

$$MRR = \frac{1}{|D|} \sum_{i=1}^{|D|} \frac{1}{Rank_i} \tag{3.8}$$

where $Rank_i$ is the position at which the most relevant (dominant) emotion for the document is ranked. $|D|$ is the total number of documents in the evaluation set.

MAP is a standard metric used in the fields of information retrieval and recommendation systems to measure the overall quality of the responses recommended or retrieved for a query. MAP of a method in predicting the order of multiple emotions connoted by a document collection D, is calculated as follows:

$$AP(d_j) = \sum_{i=1}^{k} Pr@i \times CinR@i \tag{3.9}$$

where $Pr@i$ (precision at i) is the proportion of correctly predicted emotions among the first i positions in the ranking. $CinR@i$ (change in recall at i) is $1/k$ if the emotion predicted at position i is correct, otherwise zero. Finally MAP is calculated as follows:

$$MAP = \sum_{j=1}^{|D|} \frac{AP(d_j)}{|D|} \tag{3.10}$$

3.7.3 Sentiment Score Prediction for Words/Phrases

In this task, different methods are assessed for performance in predicting the sentiment score for a set of words/phrases. Further the sentiment scores predicted are used to arrange the words/phrases in decreasing order of positivity. In this task, a standard metric Spearman's rank correlation coefficient is used for performance evaluation. The metric captures how well the predicted ranking is correlated with the ranking provided by humans. Spearman's rank correlation coefficient is calculated as follows:

$$\rho = 1 - \frac{6\sum_{i=1}^{k} {d_i}^2}{k(k^2-1)} \tag{3.11}$$

where d_i is the difference between the pair of ranks being compared and k is the size of the ranked lists. The higher the Spearman's rank correlation coefficient, the stronger is the correlation between the pair of rankings.

3.7.4 Perplexity Analysis

In this section we discuss the metric used in the proposed method for lexicon generation, in order to learn optimal language models. Perplexity is the per-word average of the probability with which a language model generates the test data, where the average taken is over the number of words in the test data. More formally given a test collection of documents $D_{e_k}^{test}$ connoting emotion e_k and a language model θ_{e_k} learnt on the training data $D_{e_k}^{train}$, the quality of the model θ_{e_k} can be estimated empirically using a standard metric Perplexity as follows:

$$Perp(D_{e_k}^{test}) = 2^{-\frac{\sum_{i=1}^{|D_{e_k}^{test}|}\sum_{j=1}^{|d_i|} logP(d_{ij}|\theta_{e_k})}{|V_{e_k}|}} \tag{3.12}$$

where V_{e_k} is the total number of words in the test data $D_{e_k}^{test}$. Perplexity measures how well the language model predicts the test (unseen) data. Therefore smaller the perplexity score, the better is the language model in predicting unseen data.

3.8 Chapter Summary

In this chapter we presented the details of the different GPELs, DSELs which are used as baselines in our comparative study. Thereafter we presented details of the baseline features used to represent text for emotion classification. We also discussed the state-of-the-art machine learning classifier, support vector machine (SVM). Finally we presented the details of the different benchmark data sets, evaluation metrics employed in this research.

Chapter 4

Generative Mixture Model for Word-Emotion Lexicon

In this chapter we present the proposed method based on Expectation Maximization to learn a word-emotion lexicon for emotion detection from text. First we formally outline the problem of emotion lexicon generation, thereafter we go into the technical details of the lexicon generation process. Finally we also illustrate the lexicon generation process on sample data.

4.1 Problem Definition

The problem essentially is to learn a word-emotion lexicon from an input document corpus, X, labelled using a pre-defined emotion set, E. Towards this objective, each subset of input documents connoting a certain emotion $e \in E$ is modelled using a unigram mixture model. Thereafter the parameters of each mixture model are estimated, to finally deduce a word-emotion lexicon.

More formally, given a corpus of documents X, with emotion labels from $E = \{e_1, \ldots, e_k\}$, the objective is to learn a word-emotion lexicon Lex, which is $|V| \times (k+1)$ matrix, where $Lex(i,j)$ is the emotional valence of the i^{th} word in vocabulary V to the j^{th} emotion in E and $Lex(i,k+1)$ corresponds to its neutral valence. The word-emotion lexicon is learnt using a set of k (one for each emotion) unigram mixture models (UMMs), each of which assumes

Notation	Description
X	Corpus of emotion labelled documents
E	Set of emotion labels
D_{e_t}	Documents labelled with emotion e_t
N	Neutral (background) language model
θ_{e_t}	Language model for emotion e_t
V	Set of unique words from documents in X
w_i	i^{th} word in the vocabulary V
Z_{w_i}	Hidden (unobserved) variable corresponding to w_i
λ_{e_t}	Mixture parameter (empirically estimated)
n	EM iteration number
$L(\theta_{e_t})$	Incomplete likelihood function
$L_{com}(\theta_{e_t})$	Complete likelihood function
$Q(\theta_{e_t};\theta_{e_t}^{(n)})$	Q-function (Expectation of the complete likelihood function)
μ	Lagrange multiplier
$c(w, d_i)$	#times word w occurs in document d_i
$Lex(i, j)$	Emotional valence between word w_i and emotion e_j
$Lex(i, k+1)$	Neutral valence for the word w_i

TABLE 4.1: Notations

that every document in X is a mixture of words connoting at least one emotion in E, and some background (neutral) words. Therefore each mixture model is a linear combination of two unigram language models, θ and N along with a mixing parameter λ. The conceptual diagram of the proposed mixture model is shown in Figure 4.1. Initial models $\theta_{e_t}^{(0)}$ and N are learnt from the training data corresponding to emotion e_t and corpus X. Mixture parameter λ_{e_t} is set empirically. The estimation of the parameters (binary hidden variables) $Z'_w s$ happens in the E-step. In the M-step parameter, θ_{e_t} is updated. This process repeats until the values of θ_{e_t} do not change significantly. In the following sections we formulate the UMM corresponding to emotion e_t and also estimate its parameters using Expectation Maximization (EM) [84, 85]. Similarly UMMs of other emotions in E can be estimated. We conclude with the UMM lexicon generation. The mathematical notations are summarized in Table 4.1.

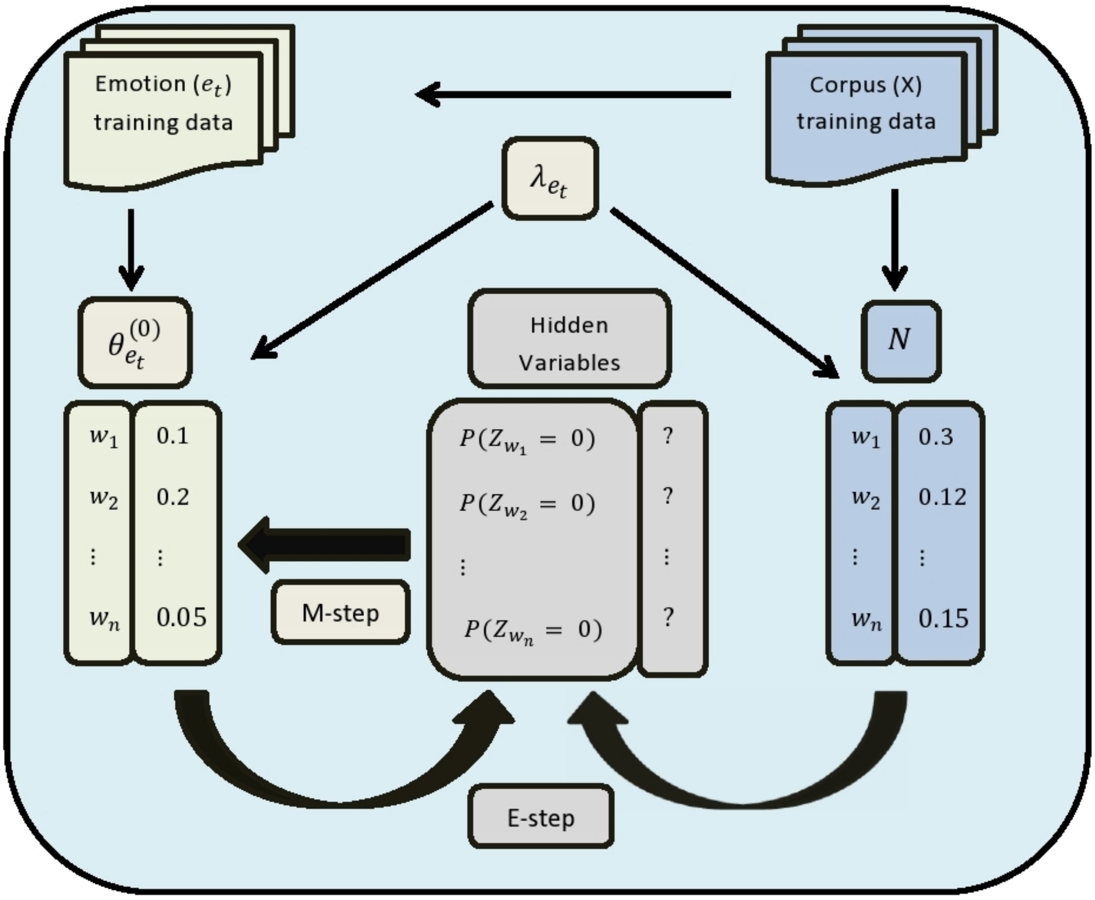


FIGURE 4.1: Visualization of the UMM generation and the EM iterative process for emotion e_t

4.2 Generative Models for Documents

We now outline two different generative models for emotion bearing documents. We start with a simple model, identify its conceptual flaws through an example, followed by a more sophisticated model for document generation.

4.2.1 Single Unigram Model

A simple model to assume for the generation of emotion (e_t) bearing documents, $D_{e_t} = \{d_1, \dots, d_m\}$, is to assume a unigram language model, θ_{e_t}, which independently generates each word w in documents from D_{e_t} as follows:

$$P(D_{e_t}|\theta_{e_t}) = \prod_{i=1}^{|D_{e_t}|} \prod_{w \in d_i} P(w|\theta_{e_t})^{c(w,d_i)} \tag{4.1}$$

where $P(w|\theta_{e_t})$ is the maximum likelihood estimate for the documents in D_{e_t}. This simple model would be reasonable if every document in D_{e_t} contain only words which bear emotion e_t. However in real-world data this is highly unlikely and documents tend to contain background (emotion-neutral) words and also other emotion words. For example consider the tweet *Sunday in Lasvegas #excited #joyous* which explicitly connotes emotion *joy*. However the word *Sunday* is evidently not indicative of *joy*. Further *Lasvegas* could connote emotions such as *Love*. Therefore it is important to have a model which accounts for such word mixtures in the documents.

4.2.2 Unigram Mixture Model

As discussed above, though a document is labelled with an emotion, not all words relate strongly to the labelled emotion. We now describe a generative model which assumes a mixture of two unigram language models to account for the aforementioned word mixtures in the documents. As mentioned before, let D_{e_t} be the documents labelled with emotion e_t, then according to the unigram mixture model documents in D_{e_t} are generated independently from a linear mixture of an emotion language model θ_{e_t} and a background language model N as follows:

$$\begin{aligned} P(D_{e_t}, Z|\theta_{e_t}) = \prod_{i=1}^{|D_{e_t}|} \prod_{w \in d_i} &[(1 - Z_w)\lambda_{e_t} P(w|\theta_{e_t}) \\ &+(Z_w)(1 - \lambda_{e_t}) P(w|N)]^{c(w,d_i)} \end{aligned} \tag{4.2}$$

Note that the above mixture model reduces to the simple language model (equation 4.1) when λ_{e_t} is 1. Thus λ_{e_t} in our case indicates the noisy (neutral and other emotion) words which occur in documents connoting emotion e_t. We show later (refer section 6.3.2) how the parameter λ_{e_t} is empirically set. Finally Z_w is the hidden/latent binary variable corresponding to word w, which indicates the mixture component (language model) that generated the word w. For each word $w \in V$ its corresponding hidden variable is defined as follows:

$$Z_w = \begin{cases} 1 & \text{if word } w \text{ is from the neutral model} \\ 0 & \text{otherwise} \end{cases}$$

The variable Z is considered to be hidden/latent since the observable data is incomplete and does not indicate how exactly each word is sampled ($P(w|\theta_{e_t})$ or $P(w|N)$) to generate the documents. We thus assume that the complete data would not only have words which generated the documents in D_{e_t} but also their corresponding values for Z. In the following sections we illustrate how the parameters (θ_{e_t}, λ_{e_t} and Z) of the mixture model are estimated.

4.2.3 Unigram mixture model for text analysis

The aforementioned unigram mixture model can be applied to problems in text analysis wherein documents occur as mixtures of words that exhibit a topic/concept and other general English terms. In particular it is more relevant to model short text (e.g. sentences, tweets etc) for detecting topics, emotions, sentiments etc. Further since real world sentiment and emotion data (e.g. opinion bearing tweets, feedback review sentences) generally compose word mixtures that are a combination strong sentiment/emotion words and other general words the unigram mixture model is more suited to model the associations between words and emotions/sentiments. Furthermore the ability of the unigram mixture model to capture the association strength between words and emotion/sentiment classes using probability scores makes it more relevant for emotion/sentiment analysis, since some words convey multiple emotions/sentiments. For example the word *accident* is associated with *fear* and *anger*.

4.3 Parameter Estimation of the Mixture Model

The objective of parameter estimation in mixture models is to find a set of parameters that maximize the probability of generating the observed data (documents). Similarly for estimating parameters of the mixture model concerning documents connoting emotion e_t, the objective is to find the parameters(θ_{e_t}, λ_{e_t} and Z) that maximize the probability of generating documents in D_{e_t}. One of the standard ways for parameter estimation is Maximum Likelihood Estimation (MLE) which observes the log-likelihood of the parameters given the data. Thereafter the parameters which maximize the log-likelihood of the data are chosen. More formally the log-likelihood of the observed data (D_{e_t}) and the complete data (D_{e_t}, Z) according to the mixture model is as follows:

$$\begin{aligned}
logL(\theta_{e_t}) &= logP(D_{e_t}|\theta_{e_t}) \\
&= \sum_{i=1}^{|D_{e_t}|} \sum_{w \in d_i} c(w, d_i) log[\lambda_{e_t} P(w|\theta_{e_t}) \\
&+ (1 - \lambda_{e_t}) P(w|N)]
\end{aligned} \tag{4.3}$$

$$\begin{aligned}
logL_{com}(\theta_{e_t}) &= logP(D_{e_t}, Z|\theta_{e_t}) \\
&= \sum_{i=1}^{|D_{e_t}|} \sum_{w \in d_i} c(w, d_i)[(1 - Z_w) log(\lambda_{e_t} P(w|\theta_{e_t})) \\
&+ (Z_w) log((1 - \lambda_{e_t}) P(w|N))]
\end{aligned} \tag{4.4}$$

Note that in equation 4.4 the sum is outside of the logarithm, since we assume that the component model used to generate each word w is known. The parameter λ_{e_t} can be estimated over the observed data D_{e_t} as follows:

$$\hat{\lambda}_{e_t} = \underset{\lambda_{e_t}}{\text{argmax}} logL(\theta_{e_t}) \tag{4.5}$$

In other words the λ_{e_t} which maximizes the log-likelihood of documents (D_{e_t}) is chosen. Essentially the parameter λ_{e_t} describes the proportion of words in the document set D_{e_t} that convey emotion e_t.We follow the same procedure to estimate λ for each emotion in the experimental data sets. (refer chapter 7). The estimation of the other parameters θ_{e_t} and Z cannot be done directly, since the MLE involves taking the derivatives of the likelihood function with respect to all unknowns (θ_{e_t}, Z) and simultaneously solving the resulting equations. This leads to a set of interlocking equations in which the solution to θ_{e_t} requires the values of Z and vice versa, thereby leading to an unsolvable equation. In such cases where a direct solution is not possible Expectation Maximization (EM) [84, 85] is applied to find the maximum likelihood estimate.

4.3.1 Expectation Maximization (EM) for parameter estimation

The central idea of EM is to maximize the probability of the complete data. EM does this iteratively by alternating between two steps (E-step and M-step). In the E-step a tight lower bound for the log-likelihood (equation 4.4) called the Q-function is calculated, which is the expectation of the complete log-likelihood function with respect to the conditional distribution of hidden variable Z given the observed data X and the current estimate of the parameter $\theta^{(n)}$:

$$Q(\theta;\theta^{(n)}) = E_{P(Z|X,\theta^{(n)})}[L_c(\theta)] = \sum_{Z} L_c(\theta)P(Z|X,\theta^{(n)})$$

The Q-function for the mixture model is as follows:

$$\begin{aligned} Q(\theta_{e_t};\theta_{e_t}^{(n)}) = & \\ & \sum_{i=1}^{|D_{e_t}|}\sum_{w\in d_i} c(w,d_i)[P(Z_w=0|D_{e_t},\theta_{e_t}^{(n)})log(\lambda_{e_t}P(w|\theta_{e_t})) \\ & + P(Z_w=1|D_{e_t},\theta_{e_t}^{(n)})log((1-\lambda_{e_t})P(w|N))] \end{aligned}$$

In the M-step a new θ = $\theta^{(n+1)}$ is computed which maximizes the Q-function that is derived in the E-step. Thus the EM algorithm is as follows:

1. Initialize $\theta^{(0)}$ randomly or heuristically
2. Iteratively improve the estimate θ by alternating between the following:
   - The E-step (expectation): Compute $Q(\theta;\theta^{(n)})$
   - The M-step (maximization): Re-estimate θ by maximizing the Q-function:

     $$\theta^{(n+1)} = \underset{\theta}{\text{argmax}} Q(\theta;\theta^{(n)})$$
3. Stop when $L(\theta)$ converges

Thus the EM algorithm iteratively expects the complete likelihood function (Q-function) and maximizes the expected Q-function, in order to re-estimate the Q-function and repeat this until the estimates do not change significantly.

4.3.2 EM steps for the mixture model

In the following sections we derive the E and M steps for the mixture model formulated over the complete data, thereafter we show how an emotion lexicon can be deduced from the mixture models defined for each emotion in E.

4.3.2.1 E-step

The major computation to be carried out in the E-step is to estimate $P(Z_w = 0|D_{e_t}, \theta_{e_t}^{(n)})$. Note that the events $P(Z_w = 0|D_{e_t}, \theta_{e_t}^{(n)})$ and $P(Z_w = 1|D_{e_t}, \theta_{e_t}^{(n)})$ are mutually exclusive and exhaustive. Therefore solving for one of them, gives the solution for the other by simply using the condition that :

$$P(Z_w = 0|D_{e_t}, \theta_{e_t}^{(n)}) + P(Z_w = 1|D_{e_t}, \theta_{e_t}^{(n)}) = 1 \tag{4.6}$$

From Bayes' theorem it follows that:

$$P(Z_w = 1|D_{e_t}, \theta_{e_t}^{(n)}) = C \times (1 - \lambda_{e_t}) \times P(w|N) \tag{4.7}$$

where C is a constant. Similarly the complementary event $P(Z_w = 0|D_{e_t}, \theta_{e_t}^{(n)})$ is:

$$P(Z_w = 0|D_{e_t}, \theta_{e_t}^{(n)}) = C \times \lambda_{e_t} \times P(w|\theta_{e_t}^{(n)}) \tag{4.8}$$

Using (4.6), (4.7) and (4.8) we have:

$$C = \frac{1}{\lambda_{e_t} \times P(w|\theta_{e_t}^{(n)}) + (1 - \lambda_{e_t}) \times P(w|N)} \tag{4.9}$$

Combining (4.8) and (4.9) gives:

$$P(Z_w = 0|D_{e_t}, \theta_{e_t}^{(n)}) = \frac{\lambda_{e_t} \times P(w|\theta_{e_t}^{(n)})}{\lambda_{e_t} \times P(w|\theta_{e_t}^{(n)}) + (1 - \lambda_{e_t}) \times P(w|N)} \tag{4.10}$$

Essentially the E-step utilizes the knowledge λ_{e_t} (i.e. the proportion of words in the document set D_{e_t} that convey emotion e_t) in order to predict whether a word w in the vocabulary V is generated by $\theta_{e_t}^{(n)}$ or N (i.e. is Z_w=0 or Z_w=1 ?)

4.3.2.2 M-step

The M-step involves maximizing the Q-function. This can be done by using a Lagrange multiplier method since we have the following constraint:

$$\sum_{w \in V} P(w|\theta_{e_t}) = 1 \tag{4.11}$$

We thus consider the auxiliary function

$$g(\theta_{e_t}) = Q(\theta_{e_t}; \theta_{e_t}^{(n)}) + \mu(1 - \sum_{w \in V} P(w|\theta_{e_t})) \tag{4.12}$$

where μ is the Lagrange multiplier. Computing the first-order partial derivative of $g(\theta_{e_t})$ with respect to the parameter variable $P(w|\theta_{e_t})$ and equating to zero we get:

$$\mu = \frac{\sum_{i=1}^{|D_{e_t}|} \sum_{w \in d_i} P(Z_w = 0|D_{e_t}, \theta_{et}^{(n)})}{P(w|\theta_{e_t})} \tag{4.13}$$

$$\Rightarrow P(w|\theta_{e_t}) = \frac{\sum_{i=1}^{|D_{e_t}|} P(Z_w = 0|D_{e_t}, \theta_{et}^{(n)}) c(w, d_i)}{\mu} \tag{4.14}$$

Now using the constraint that $\sum_{w \in V} P(w|\theta_{e_t}) = 1$, we get

$$\sum_{w \in V} P(w|\theta_{e_t}) = \frac{\sum_{w \in V} \sum_{i=1}^{|D_{e_t}|} P(Z_w = 0|D_{e_t}, \theta_{et}^{(n)}) c(w, d_i)}{\mu} = 1 \tag{4.15}$$

$$\Rightarrow \mu = \sum_{w \in V} \sum_{i=1}^{|D_{e_t}|} P(Z_w = 0|D_{e_t}, \theta_{et}^{(n)}) c(w, d_i) \tag{4.16}$$

The above equation is a result of a simple cross multiplication between the two components on the right hand side of equation 4.15. Using (4.15) and (4.16) we get:

$$P(w|\theta_{e_t}) = \frac{\sum_{i=1}^{|D_{e_t}|} P(Z_w = 0|D_{e_t}, \theta_{et}^{(n)}) c(w, d_i)}{\sum_{w \in V} \sum_{i=1}^{|D_{e_t}|} P(Z_w = 0|D_{e_t}, \theta_{et}^{(n)}) c(w, d_i)} \tag{4.17}$$

The M-step receives inputs from the E-step estimates to update the probabilistic mass for a word w with respect to an emotion e_t. These updated probability values from the M-step form as inputs for the E-step in the next EM iteration, until convergence. In summary we have the following update formulas for the mixture model

E-step:

$$P(Z_w = 0|D_{e_t}, \theta_{e_t}^{(n)}) = \frac{\lambda_{e_t} P(w|\theta_{e_t}^{(n)})}{\lambda_{e_t} P(w|\theta_{e_t}^{(n)}) + (1-\lambda_{e_t}) P(w|N)} \tag{4.18}$$

M-step:

$$P(w|\theta_{\theta_{e_t}}^{(n+1)}) = \frac{\sum_{i=1}^{|D_{e_t}|} P(Z_w = 0|D_{e_t}, \theta_{e_t}^{(n)}) c(w, d_i)}{\sum_{w \in V} \sum_{i=1}^{|D_{e_t}|} P(Z_w = 0|D_{e_t}, \theta_{e_t}^{(n)}) c(w, d_i)} \tag{4.19}$$

4.3.3 EM Initialization

EM algorithm can be started by either initializing the set of parameters (θ) and then conducting an E-step or by starting with a set of initial estimates for the hidden variables (Z) and then conducting an M-step. In our case since the mixture model is applied on labelled or weakly-labelled data, initial values for the parameters (θ) can be chosen heuristically, before alternating between E-step and M-step till convergence. The initial language model $\theta_{e_t}^{(0)}$ for emotion e_t is defined as follows:

$$P(w_i|\theta_{e_t}^{(0)}) = \frac{f(w_i, D_{e_t})}{\sum_{w \in V} f(w, D_{e_t})} \tag{4.20}$$

where $f(w_i, D_{e_t})$ is the frequency of the i^{th} word in V in training documents labelled with emotion e_t. The background (neutral) language model is defined as follows:

$$P(w_i|N) = \frac{f(w_i, X)}{\sum_{w \in V} f(w, X)} \tag{4.21}$$

where $f(w_i, X)$ is the training corpus frequency for word w_i. We believe it is reasonable to use the corpus X to model the words which do not convey the dominant emotion in the document, since words occurring in multiple class (emotion) documents tend to have higher corpus frequencies and hence a higher score in the neutral model.

Words	Anger	Fear	Joy	Sadness	Surprise	Neutral
:)	0.056	0.085	0.533	0.062	0.072	0.192
good!!	0.074	0.109	0.305	0.236	0.093	0.183
#arrogant	0.332	0.173	0.057	0.131	0.150	0.157

TABLE 4.2: A sample of the UMM word-emotion lexicon

4.4 Lexicon Generation

EM is used to estimate the parameters of the k mixture models corresponding to the emotions in E as illustrated earlier. The objective of a word-emotion lexicon is to capture the word level emotion characteristics and to be able to quantify the strength of association between a word and a range of emotions. Since an emotion language model (θ_{e_i}) captures the association between a word w and emotion e_i in the form of a likelihood estimate $P(w|e_i)$, we can generate a word-emotion lexicon using the k emotion language models and the background model N as follows:

$$Lex^{(n)}(w_i, \theta_{e_j}) = \frac{P(w_i|\theta_{e_j}^{(n)})}{\sum_{t=1}^{k}[P(w_i|\theta_{et}^{(n)})] + P(w_i|N)} \tag{4.22}$$

$$Lex^{(n)}(w_i, N) = \frac{P(w_i|N)}{\sum_{t=1}^{k}[P(w_i|\theta_{et}^{(n)})] + P(w_i|N)} \tag{4.23}$$

where k is the number of emotions in the corpus, and $Lex^{(n)}$ is a $|V| \times (k+1)$ matrix generated after the n^{th} EM iteration. Thus each word in the lexicon has a corresponding probability distribution over the k emotion classes and the neutral class. A sample of the UMM lexicon generated on blogs is shown in Table 4.2.

The key algorithmic steps involved in generating a word-emotion lexicon using expectation maximization are shown in Table 4.3. Observe that the emotion language models are replaced with the sentiment language models when expectation maximization is applied to induce a sentiment lexicon. We will describe the mathematical formulation for inducing a sentiment lexicon using expectation maximization in Chapter 6.

States	EM algorithm
Input	Training data T
Output	Word-emotion lexicon L
Initialisation	Learn the initial language models
Convergence	While not converged or #iterations less than δ a threshold
E-step	Estimate $P(Z_w = 0\|D_{e_t}, \theta_{e_t}^{(n)})$ using the current estimates of the emotional language model $(P(w\|\theta_{e_t}^{(n)}))$, neutral language model $(P(w\|N))$ and the parameter λ_{e_t}
M-step	Re-estimate the emotional language models $(P(w\|\theta_{e_t}^{(n+1)}))$ using the estimates obtained from the E-step
Lexicon Induction	Induce a word-emotion lexicon using the final estimates for the emotion language model $(P(w\|\theta_{e_t}^{(n)}))$ and the neutral language model $(P(w\|N))$

TABLE 4.3: EM algorithm steps for generating a word-emotion lexicon

4.5 Lexicon Generation: A Walk through Example

In this section, we illustrate the various steps involved in the lexicon generation using the proposed method, with the help of sample Twitter data. For ease of explanation, we consider only data from two emotion classes to explain the different steps of the lexicon generation. The data used to train the lexicon for the two emotion classes, *anger and joy* is shown in Tables 4.4 and 4.5.

4.5.1 Initial language model generation

The initial language models corresponding to the emotion classes *anger* and *joy* and the background language model are generated according to equations 4.20 and 4.21 in section 4.3.3. At the end of this step, the initial language models $\theta_{Anger}^{(0)}$, $\theta_{Joy}^{(0)}$ and the background model N are generated. A sample of these language models on the toy data set (*anger*, *joy* documents) is shown in Table 4.6.

Anger training set	Anger validation set
1. wish people would just be honest...	1. ugh.i hate you
2.this kids are so immature #badmood	2. grr my phone isn't getting texts
3. restless nights cuz of a bad shoulder	3. really wish my ears would pop
4.when someone knows their funny but then they try too hard all the time	4. wish i knew what it was like to not be treated like shit #fuckeveryone
5. I don't get this cold one day im good the next i feel like shit then better again	5. if this drunk ass man dnt stfu imma murk.. he kp walkin back and forth tlkin wreckless...
6. ugh.i have too continue my great turkey weekend with my dads gf and her three weird kids.	6. off to memphis. and not by choice this time. gotta love mandatory work drug tests.
7. I hate when i drive at night and people coming towards me or behind me have their brights on.	

TABLE 4.4: Anger documents

Joy training set	Joy validation set
1. yay got the job at jc penney!!!(:	1. my grades are amazing!! (:
2. one week til christmas!	2. my life is great
3. my life is so amazingly awesome	3. i hate packing -.- but i love what i'm packing for!!! =)
4. can't wait to start planning our trip to mardi gras 2012!!!:)	4. i can't sleep and now i have nothing to do 2.5hours til work
5. i is healthy, today. and yesterday. and the day before. #ignorethis	5. going to colombia this summer for the first time :) hahaaa!
6. you've been on my mind a lot lately, and i kinda like it actually.	6.the pants are off and my room is finally beginning to cool down. the early signs of being comfortable.
7. goodmorning texts and compliments make my day so much better! #goodlife	
8. christmas movies and dominos with babe ;) then heather is gonnnnna be home yay!	

TABLE 4.5: Joy documents

Words	$\theta^{(0)}_{Anger}$	$\theta^{(0)}_{Joy}$	N
hate	0.0086	0.0032	0.0054
yay	0.0045	0.0133	0.0089
⋮	⋮	⋮	⋮
time	0.0090	0.0088	0.0089
#goodlife	0.0045	0.0088	0.0067
shit	0.0090	0.0044	0.0067
⋮	⋮	⋮	⋮

TABLE 4.6: Initial language models

4.5.2 Parameter Estimation

In this section, we illustrate the estimation process for the parameters λ_{Anger}, λ_{Joy}, θ_{Anger}, θ_{Joy} and Z. In order to estimate the optimal values for λ_{Anger} and λ_{Joy}, we observe the likelihood of unseen data shown in Tables 4.4 and 4.5 according to the formulations in equations 4.4 and 4.5. In this research, we experimented with 11 different values [0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0] of λ and selected the one, which maximizes the log-likelihood of unseen data. As mentioned before λ is inversely proportional to the noise in the documents. Therefore, if λ for an emotion is closer to 1, its documents are emotion-rich.

Tables 4.7 and 4.8 capture the log likelihood values for different values of λ across two EM iterations. It is evident from these tables that the initial language models $\theta^{(0)}_{Anger}$ and $\theta^{(0)}_{Joy}$ best generate the unseen data for λ values 0.9 and 0.8 respectively. This shows that the assumption of real world data to be a mixture of emotion-bearing and emotion-neutral words is valid. In the E-step (refer equation 6), the optimum values for λ are substituted to estimate the probability values for the hidden variables Z being zero or not. Thereafter those probability values are used in the M-step (refer equation 4.19) to estimate the language models $\theta^{(1)}_{Anger}$, $\theta^{(1)}_{Joy}$ and similarly $\theta^{(2)}_{Anger}$ and $\theta^{(2)}_{Joy}$. Table 4.9 shows the updated language models after two EM iterations. Since, the proposed method for lexicon generation is applied on a tiny data set, convergence of the EM iterations happened very quickly. However on a real world data set, it is expected to see more iterations before convergence. We illustrate our findings on real world data sets in chapter 7.

$\lambda_{\theta_{Anger}}$	Likelihood for $\theta^{(0)}_{Anger}$	Likelihood for $\theta^{(1)}_{Anger}$	Likelihood for $\theta^{(2)}_{Anger}$
0.0	-167.4904	-167.4904	-167.4904
0.1	-167.1203	-167.0863	-167.0821
0.2	-166.7890	-166.7288	-166.7215
0.3	-166.4965	-166.4178	-166.4085
0.4	-166.2432	-166.1540	-166.1441
0.5	-166.0300	-165.9389	-165.9296
0.6	-165.8586	-165.7749	-165.7678
0.7	-165.7313	-165.6657	-165.6627
0.8	-165.6516	-165.6471	-165.6307
0.9	-165.6509	-165.6375	-165.6248
1.0	-165.7681	-165.7398	-165.6583

TABLE 4.7: Anger Log Likelihood Estimates

$\lambda_{\theta_{Joy}}$	Likelihood for $\theta^{(0)}_{Joy}$	Likelihood for $\theta^{(1)}_{Joy}$	Likelihood for $\theta^{(2)}_{Joy}$
0.0	-178.7519	-178.7519	-178.7519
0.1	-178.2375	-178.1898	-178.1842
0.2	-177.7716	-177.6859	-177.6759
0.3	-177.3524	-177.2381	-177.2249
0.4	-176.9788	-176.8449	-176.8296
0.5	-176.6503	-176.5058	-176.4896
0.6	-176.3665	-176.2207	-176.2048
0.7	-176.2779	-176.0805	-176.0061
0.8	-175.6939	-175.6485	-175.6486
0.9	-175.7898	-175.7013	-175.6947
1.0	-175.9352	-175.8165	-175.8052

TABLE 4.8: Joy Log Likelihood Estimates

Words	$\theta^{(1)}_{Anger}$	$\theta^{(1)}_{Joy}$	$\theta^{(2)}_{Anger}$	$\theta^{(2)}_{Joy}$	N
hate	0.0087	0.0031	0.0087	0.0030	0.0054
yay	0.0040	0.0137	0.0040	0.0138	0.0089
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
time	0.0090	0.0088	0.0090	0.0088	0.0089
#goodlife	0.0042	0.0091	0.0042	0.0091	0.0067
shit	0.0092	0.0042	0.0092	0.0041	0.0067
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$

TABLE 4.9: Emotion language models over two EM iterations

4.5.3 Lexicon Generation

Lexicon generation is done using the optimal language models for the emotions *anger, joy* i.e. $\theta^{(2)}_{Anger}$ and $\theta^{(2)}_{Joy}$ and the background (neutral) language model N as shown in equations 4.22 and 4.23. Table 4.10 shows the emotion lexicon obtained by row normalizing the language models, $\theta^{(2)}_{Anger}$, $\theta^{(2)}_{Joy}$ and N. Observe that emotion-rich words such as *hate* and *shit* are assigned high scores with emotion *anger*, where as words such as *yay* and *#goodlife* are assigned high scores with emotion *joy*. Further emotion-neutral word such as *time* is penalized by the background language model, since it evidently conveys neither *anger* nor *joy*. The proposed mixture model is able to capture such word-mixtures present in the data and quantify their emotionality accordingly. We expect such ability to model emotionality at word-level is important for performance in different emotion detection tasks, which are discussed later (refer chapter 7).

4.6 Chapter Summary

In this chapter, first we outlined the problem of lexicon generation. Second, we introduced a domain specific emotion lexicon (DSEL) generation method which can extract a word-emotion association lexicon from a corpus of emotion labelled text. The DSEL is a result of a novel unigram mixture model (UMM), which models text as a mixture of emotion-rich and neutral words. It is unique in its ability to model emotionality as well as neutrality of words. The parameters of the proposed UMM are estimated using a popular technique of expectation maximization (EM).

Words	Anger	Joy	Neutral
hate	0.5087	0.1754	0.3158
yay	0.1502	0.5156	0.3340
⋮	⋮	⋮	⋮
time	0.3357	0.3308	0.3333
#goodlife	0.2120	0.4540	0.3338
shit	0.4591	0.2080	0.3327
⋮	⋮	⋮	⋮

TABLE 4.10: UMM Emotion Lexicon

Finally we illustrate each step of the DSEL generation on a sample data from Twitter. Evaluation of the UMM emotion lexicon is presented in Chapter 7 through different emotion detection tasks: *word-emotion classification, document-emotion ranking and document-emotion classification*. We evaluate the quality of the proposed UMM lexicon in comparison with existing GPELs and DSELs generated using state-of-the-art methods such as point-wise mutual information (PMI) and latent Dirichlet allocation (LDA).

Chapter 5

Lexicon-based Emotion Feature Extraction

In this chapter we first motivate the need for lexicon based feature extraction for emotion classification. Thereafter, we introduce novel feature extraction methods to harness the emotion rich knowledge being captured by our Domain specific emotion lexicon (DSEL). The proposed features are used to represent documents along emotion concepts in order to classify them into emotion classes using machine learning. Further it is also possible to extract the proposed features from existing general purpose emotion lexicons (GPELs) and also other DSELs. The unique ability of the proposed DSEL to model emotionality and neutrality of words, is expected to enrich the text representations proposed, thereby leading to performance improvements in emotion classification. Finally we also introduce hybrid features for emotion classification, obtained by combining lexicon-based features and other standard features such as bag-of-words (BoW) used in the literature for emotion classification. We present the details of each of the text representations, first through mathematical formulations, followed by visual explanations using matrix-vector notations.

5.1 Text Representation for Emotion Classification

Representation of text documents is a crucial step in machine learning approaches for text classification. A popular representation involves refining the BoW or n-grams feature vector, so that a subset of words are chosen using a selection metric to represent a text document [94]; this is normally referred to as feature selection. Feature engineering, on the other hand, is about building a set of new features rather than selecting a subset of words. Such features could be frequency of higher-level concepts such as *topics* [95], or may use semantic representations derived from an ontology [96]. More specialized tasks call for more fine-tuned feature representations; for example, the length of contiguous upper-case character sequences is found to be a useful feature for spam filtering[1] whereas the number of lower-cased words resulted in performance improvements in SMS filtering [97]. Author identification is another area where fine-tuned features such as stylemarkers and short-words (e.g. if, is etc.) have enhanced classification accuracy [98]. Document-emotion classification, being as much or even more specialized than the above tasks,also requires fine-tuned features.

Emotion classification of text requires careful modelling, since words associate with different emotions in different contexts with varying levels of magnitude making the identification of words for document representation more challenging. For example, in a sentence such as *beautiful morning #amazing* the word *beautiful* could be associated moderately with emotions such as *joy* and *love*, *amazing* could be associated strongly with emotion *joy* and *morning* could be weakly associated with emotion *joy*. Such word-emotion associations are usually captured by emotion lexicons. Existing general purpose emotion lexicons (GPELs) such as WordNet-Affect (WNA) [4], EmoSenticNet (ESN) [70] and NRC word-emotion lexicon [5], which are hand crafted, associate between words and emotions identified by Ekman and Plutchik. Emotion features extracted using the knowledge of the GPELs, when combined with traditional BoW features improved emotion classification significantly [30, 64].

However GPELs poorly model the context in which words convey emotions. For example *Glee* might normally connote *joy*, but would need to be assumed neutral in the context of a document corpus talking about the television series with the same name. Further, *unfair* may be associated with *anger* despite being more dominant in *sadness* related documents; the crisp binary

[1] http://archive.ics.uci.edu/ml/datasets/Spambase

memberships of words in GPELs do not allow to capture such fuzzy memberships of words to emotion classes, thereby making them limitedly effective for feature extraction. Accordingly, recent efforts in emotion detection focused on learning domain specific lexicons [46, 99] and also utilizing them for emotion feature extraction [5, 48]. However the emotion features extracted were limited to simple emotion word counts in a document using the lexicon, which, while being simple, do not exploit the knowledge of the lexicon in its entirety. As mentioned in the related work section, previous research suggests that lexicon based features improve emotion classification. Therefore in this research we further explore the role of lexicon based features for effective emotion classification by extracting novel features utilizing the knowledge of the proposed DSEL.

5.2 Lexicon-based Features for Emotion Classification of Text

In this section we explore how the knowledge of a DSEL can be utilized to extract a range of features relevant for emotion classification. In particular we are interested in exploring the knowledge captured by the proposed UMM lexicon along with other baseline DSELs to extract text representations for effective classification using machine learning. The performance contributions of the feature vectors learnt using the knowledge of the proposed DSEL is validated through emotion classification tasks on text from different domains. Observe that all the lexicon based feature vectors proposed in this research are of length $|E|$, where $|E|$ is the number of emotion classes in a data set. This make the proposed representations dense and continuous as opposed to the sparse and high dimensional representations like BoW. We consider the following features to represent documents:

1. Total Emotion Count (TEC) [48]: This feature captures the number of words in a document that associate with an emotion. Given a document d, its corresponding feature vector is denoted by d_{TEC}. The feature value for the j^{th} emotion is computed as follows:

$$d_{TEC}[e_j] = \sum_{w \in d} I(e_j = \arg\max_k Lex(w,k)) \times count(w,d) \tag{5.1}$$

$I(.)$ is an indicator function and is set to 1 or 0 when the argument is true or false respectively. $count(w, d)$ is the number of occurrences of word w in document d. Note that TEC only captures the popular emotion context of a word suggested by the lexicon (i.e., emotion with highest score in the lexicon). However not all words associate with just a single emotion. For example, even if the word *beautiful* may be associated moderately with both the emotions *joy* and *love*, the TEC emotion feature would force the word to contribute a count of 1 towards either of these emotions (depending on the scores from the lexicon Lex) and 0 towards the other. Therefore it is important to develop features that incorporate the relations between a word and multiple emotions.

2. Total Emotion Intensity (TEI): This is the sum of the emotion intensity scores of words present in a document. Unlike the coarse integer counts in TEC features, here word-level emotion intensity scores offered by a DSEL are used to capture the emotional orientation of documents along multiple emotion concepts (classes). Accordingly d_{TEI} is the feature vector corresponding to a document d. The feature value for the j^{th} emotion is computed as follows:

$$d_{TEI}[e_j] = \sum_{w \in d} Lex(w, e_j) \times count(w, d) \tag{5.2}$$

The additional ability of TEI over TEC is that it can potentially discriminate between documents connoting emotionality with varying intensity. For example, *fantastic* is a stronger indicator of a positive emotion compared to the word *good*. Therefore it is useful to capture such information to classify documents into emotion classes. Since the DSEL captures domain level expressions that convey emotions, the emotion intensity of such expressions can be easily aggregated to the document level to model the emotion intensity of the document.

3. Max Emotion Intensity (MEI): Research in Sentiment analysis suggest that high sentiment-bearing terms are indicative of sentiment class of the document regardless of the average score for the document [78]. For example in the sentence, *the food, service and the prices were all brilliant at the Thai place*, *brilliant* is a strong sentiment bearing word, compared to the other words. Therefore the sentence, can be classified as positive based on just the sentiment information of the word *brilliant*. Similarly in the case of emotion detection, modelling for strong emotion words is expected to be effective. Therefore we consider the

intensity score of the highest emotion-bearing word in the given document to learn document representations. More formally, given a document d, and its corresponding feature vector d_{MEI}, the feature value for the j^{th} emotion is computed as follows:

$$d_{MEI}[e_j] = \arg\max_{w \in d} Lex(w, j) \tag{5.3}$$

4. Graded Emotion Count (GEC): We extend the idea of utilizing high intensity emotion words to extract document representations by developing variants of TEC and TEI. Both TEC and TEI consider all the words in a document regardless of the intensity with which they convey an emotion. However it is useful to understand the impact of high intensity words on emotion classification. GEC is similar in principle to TEC, except that it only captures the number of words in a document that associate with an emotion and over a threshold value δ. Since our proposed DSEL quantifies the association between each word and the set of emotions in the form of a probability distribution, the intensity scores always lie in the interval [0, 1]. We divide this interval into 4 quartiles [0, 0.25), [0.25, 0.5), [0.5, 0.75) and [0.75, 1] respectively. Further we use the three values 0.25, 0.5 and 0.75 as threshold δ in our experiments. The GEC features extracted using the DSELs are for the above three thresholds. Given a document d, and its corresponding feature vector d_{GEC}, the feature value for the j^{th} emotion is computed as follows:

$$d_{GEC}[e_j] = \sum_{\substack{w \in d \\ Lex(w,j) \geq \delta}} I(e_j = \arg\max_{k} Lex(w, k)) \times count(w, d) \tag{5.4}$$

5. Graded Emotion Intensity (GEI): Similar to GEC, we develop a variant of TEI, GEI which is the sum of intensity scores of words in a document and over a threshold δ. The thresholds mentioned earlier are used for extracting GEI features using DSELs. Given a document d, and its corresponding feature vector d_{GEI}, the feature value for the j^{th} emotion is computed as follows:

$$d_{GEI}[e_j] = \sum_{\substack{w \in d \\ Lex(w,j) \geq \delta}} Lex(w, e_j) \times count(w, d) \tag{5.5}$$

5.3 Hybrid Features for Emotion Classification of Text

A hybrid feature vector H is a $K + E$ dimensional feature vector obtained by combining a K dimensional baseline feature vector (refer section 3.3 in chapter 3) and a E dimensional lexicon based feature vector. In this research, we assess the performance of each type of baseline features (n-grams, part-of-speech features and contextual features such as number of elongated words, number of capitalized words, negation features etc) and identify the best performing features on each of the datasets. Further we also identify the best performing lexicon based features. Thereafter we combine these best performing features (baseline and lexicon-based) to construct the hybrid feature vectors. We expect the non-lexicon based features to be useful for emotion modelling, in domains where there is inadequate data to learn DSELs and extract lexicon-based features. Further in domains where there is adequate data to learn DSELs, the combination of lexicon-based and non-lexicon features is expected to boost performance in emotion classification. We present the details about our findings with using different text representations for emotion classification in chapter 7.

5.4 Visualizing Emotion Feature Vectors

In this section we explain visually the details of the different lexicon based text representations proposed in the previous section generated using the proposed UMM lexicon ($ElLex$). Similarly the text representations using other DSELs can be visualized. We use matrix-vector algebra to illustrate the feature vector construction. We believe this analysis of the feature vectors gives a deeper understanding of the knowledge captured by each document representation. Also it will further help in the performance analysis of machine learning approaches for emotion text classification. We consider the following toy example data to illustrate the different feature extraction strategies discussed in the previous section.

5.4.1 Sample Data

Let $D = \{d_1, d_2, d_3, d_4\}$ be the four documents, let $V = \{w_1, w_2, w_3, w_4, w_5, w_6\}$ be the vocabulary that composes the documents in D. Let $E = \{e_1, e_2, e_3, e_4, e_5\}$ be the predefined emotion classes. Before explaining the feature extraction process we define the emotion lexicon $ELex$ whose knowledge is utilized through different feature extraction methods to learn document representations. $ELex$ originally is $|V| \times |E| + 1$ matrix, where the first $|E|$ columns correspond to the emotions in $|E|$ and the last column represents neutrality. We consider the first $|E|$ columns of the matrix $ELex$ and re-normalize the rows before applying the lexicon for feature extraction. Observe that the neutrality column for each word captures probability mass that is proportional to its entropy. Therefore we expect the contribution of neutral words (i.e. words that occur near uniformly across the emotion classes) to be reduced in the feature weights, thereby causing less confusion to a machine learning classifier to decipher class boundaries. We now define the emotion lexicon $ELex$:

$$ELex = \begin{array}{cc} & \begin{array}{ccccc} e_1 & e_2 & e_3 & e_4 & e_5 \end{array} \\ \begin{array}{c} w_1 \\ w_2 \\ w_3 \\ w_4 \\ w_5 \\ w_6 \end{array} & \begin{pmatrix} 0.1510 & 0.2407 & 0.2355 & 0.1702 & 0.2024 \\ 0.1172 & 0.4422 & 0.1453 & 0.1336 & 0.1615 \\ 0.0475 & 0.0118 & 0.9255 & 0.0063 & 0.0086 \\ 0.1277 & 0.2029 & 0.0420 & 0.4564 & 0.1708 \\ 0.4288 & 0.2070 & 0.0428 & 0.1468 & 0.1743 \\ 0.0872 & 0.1448 & 0.0260 & 0.2304 & 0.5114 \end{pmatrix} \end{array} \tag{5.6}$$

In the following sections we demonstrate how the knowledge of the above lexicon is utilized through different feature extraction methods to construct document level representations. Essentially the matrix $ELex$ is transformed differently by each feature extraction method and combined with the word-document frequency matrix WDF defined below to obtain the feature vectors for the documents. WDF for the sample data is defined as follows:

$$WDF = \begin{pmatrix} & w_1 & w_2 & w_3 & w_4 & w_5 & w_6 \\ d_1 & 2 & 0 & 1 & 1 & 2 & 3 \\ d_2 & 1 & 3 & 2 & 1 & 0 & 2 \\ d_3 & 1 & 2 & 2 & 0 & 3 & 1 \\ d_4 & 3 & 0 & 2 & 1 & 1 & 2 \end{pmatrix} \tag{5.7}$$

Observe that the dimensions of WDF and $ELex$ are $|D| \times |V|$ and $|V| \times |E|$ respectively. Therefore all the resultant feature vectors obtained using WDF and $ELex$ are of the dimension $|D| \times |E|$. Also for any arbitrary document the feature vector is of the dimension $1 \times |E|$.

5.4.2 Visualizing Total Emotion Count (TEC)

As illustrated in the section 5.2 TEC feature representation for a document captures the number of words per emotion contained within the document. In order to achieve this TEC fist transforms the probabilistic word-emotion distributions in $ELex$ into binary word-vectors, wherein the emotion with the highest score for a word is assigned a value of 1, otherwise 0. The $ELex$

matrix transformed into binary word-vectors is as follows:

$$TEC_{ELex} = \begin{array}{c} \\ w_1 \\ w_2 \\ w_3 \\ w_4 \\ w_5 \\ w_6 \end{array} \begin{array}{c} \begin{array}{ccccc} e_1 & e_2 & e_3 & e_4 & e_5 \end{array} \\ \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \end{array} \tag{5.8}$$

Thereafter the word-document frequencies captured in the WDF matrix are combined with the matrix TEC_{ELex} to obtain the TEC feature vectors for documents in D. For example the TEC feature vector for document d_1, i.e. d_{1TEC} is obtained by applying the transpose of its corresponding frequency vector from WDF, i.e. <2, 0, 1, 1, 2, 3> (C) as a multiplication filter across the columns of the matrix TEC_{ELex}, followed by the sum of resultant vectors. More visually the scalar multiplication of C^T across columns of TEC_{ELex} and the summation of the resultant vectors is:

$$d_{1\,TEC} = \left(\begin{array}{cccccc} & e_1 & e_2 & e_3 & e_4 & e_5 \\ w_1 & 0 & 2 & 0 & 0 & 0 \\ & & & + & & \\ w_2 & 0 & 0 & 0 & 0 & 0 \\ & & & + & & \\ w_3 & 0 & 0 & 1 & 0 & 0 \\ & & & + & & \\ w_4 & 0 & 0 & 0 & 1 & 0 \\ & & & + & & \\ w_5 & 2 & 0 & 0 & 0 & 0 \\ & & & + & & \\ w_6 & 0 & 0 & 0 & 0 & 3 \end{array}\right) \tag{5.9}$$

(5.10)

$$= (2, 2, 1, 1, 3) \tag{5.11}$$

5.4.3 Visualizing Total Emotion Intensity (TEI)

TEI unlike TEC utilizes the emotion intensity scores to learn document representations that not only capture the emotional orientation of the document but also quantify it. In order to achieve this TEI combines the emotion lexicon $ELex$ and the word-document frequencies captured in the WDF matrix to obtain the TEI feature vectors for documents in D. For example the TEI feature vector for document d_2, i.e. d_{2TEI} is obtained by applying the transpose of its corresponding frequency vector from WDF, i.e. <1, 2, 2, 0, 3, 1> (C) as a multiplication filter across the columns of the matrix $ELex$, followed by the sum of resultant vectors. More visually the scalar multiplication of C^T across columns of $ELex$ and the summation of the resultant vectors is:

$$
d_{2\,TEI} = \begin{array}{cc}
 & \begin{array}{ccccc} e_1 & e_2 & e_3 & e_4 & e_5 \end{array} \\
\begin{array}{c} w_1 \\ \\ w_2 \\ \\ w_3 \\ \\ w_4 \\ \\ w_5 \\ \\ w_6 \end{array} &
\left(\begin{array}{ccccc}
0.1510 & 0.2407 & 0.2355 & 0.1702 & 0.2024 \\
 & & + & & \\
0.2344 & 0.8844 & 0.2906 & 0.2673 & 0.3231 \\
 & & + & & \\
0.0951 & 0.0237 & 1.8510 & 0.0127 & 0.0172 \\
 & & + & & \\
0 & 0 & 0 & 0 & 0 \\
 & & + & & \\
1.2866 & 0.6212 & 0.1285 & 0.4405 & 0.5230 \\
 & & + & & \\
0.0872 & 0.1448 & 0.0260 & 0.2304 & 0.5114
\end{array}\right)
\end{array}
\tag{5.12}
$$

$$
\tag{5.13}
$$

$$
= (1.8544, 1.9149, 2.5318, 1.1213, 1.577) \tag{5.14}
$$

5.4.4 Visualizing Max Emotion Intensity (MEI)

As illustrated in the section 5.2 MEI feature representation for a document captures the emotion association between a document and emotion, through the word that has the maximum emotion intensity. In other words MEI applies a max operation on each of the columns (i.e. for each emotion) in $ELex$ in order to identify the strongest emotion bearing word in a document. The WDF matrix is used to identify whether or not a word is present in a document in order to apply the max operation. Thereafter the max word-emotion intensities obtained for each emotion are used to form the MEI feature vectors for documents in D. For example the MEI feature vector for document d_3, i.e. d_{3MEI} is obtained first by a look up into the WDF to obtain the words that compose it i.e. w_1, w_2, w_3, w_5 and w_6 (C). Now the corresponding emotion vectors are selected from $ELex$ as shown below to apply the max operation on each column to finally obtain the feature vector d_{3MEI}.

$$d_{3\,MEI} = \left(\begin{array}{cccccc} & e_1 & e_2 & e_3 & e_4 & e_5 \\ w_1 & 0.1510 & 0.2407 & 0.2355 & 0.1702 & 0.2024 \\ w_2 & 0.1172 & 0.4422 & 0.1453 & 0.1336 & 0.1615 \\ w_3 & 0.0475 & 0.0118 & 0.9255 & 0.0063 & 0.0086 \\ w_5 & 0.4288 & 0.2070 & 0.0428 & 0.1468 & 0.1743 \\ w_6 & 0.0872 & 0.1448 & 0.0260 & 0.2304 & 0.5114 \end{array} \right) \tag{5.15}$$

$$\tag{5.16}$$

$$= (0.4288, 0.4422, 0.9255, 0.2304, 0.5114) \tag{5.17}$$

5.4.5 Visualizing Graded Emotion Count (GEC)

Unlike TEC, GEC considers only words with emotion intensity over the threshold δ. In order to achieve this GEC fist transforms the probabilistic word-emotion distributions in $ELex$ into binary word-vectors, wherein the emotion with the highest score above or equal to the threshold δ for a word is assigned a value of 1, otherwise 0. The $ELex$ matrix transformed into binary word-vectors is as follows considering the threshold δ = 0.25

$$GEC_{ELex} = \begin{pmatrix} & e_1 & e_2 & e_3 & e_4 & e_5 \\ w_1 & 0 & 0 & 0 & 0 & 0 \\ w_2 & 0 & 1 & 0 & 0 & 0 \\ w_3 & 0 & 0 & 1 & 0 & 0 \\ w_4 & 0 & 0 & 0 & 1 & 0 \\ w_5 & 1 & 0 & 0 & 0 & 0 \\ w_6 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{5.18}$$

Thereafter the word-document frequencies captured in the WDF matrix are combined with the matrix GEC_{ELex} to obtain the GEC feature vectors for documents in D. For example the GEC feature vector for document d_1, i.e. d_{1GEC} is obtained by applying the transpose of its corresponding frequency vector from WDF, i.e. <2, 0, 1, 1, 2, 3> (C) as a multiplication filter across the columns of the matrix GEC_{ELex}, followed by the sum of resultant vectors. More visually the scalar multiplication of C^T across columns of GEC_{ELex} and the summation of the resultant vectors is:

$$d_{1\,GEC} = \left(\begin{array}{cccccc} & e_1 & e_2 & e_3 & e_4 & e_5 \\ w_1 & 0 & 0 & 0 & 0 & 0 \\ & & & + & & \\ w_2 & 0 & 0 & 0 & 0 & 0 \\ & & & + & & \\ w_3 & 0 & 0 & 1 & 0 & 0 \\ & & & + & & \\ w_4 & 0 & 0 & 0 & 1 & 0 \\ & & & + & & \\ w_5 & 2 & 0 & 0 & 0 & 0 \\ & & & + & & \\ w_6 & 0 & 0 & 0 & 0 & 3 \end{array}\right) \tag{5.19}$$

$$\tag{5.20}$$

$$= (2, 0, 1, 1, 3) \tag{5.21}$$

5.4.6 Visualizing Graded Emotion Intensity (GEI)

GEI similar to TEI utilizes the emotion intensity scores to learn document representations that not only capture the emotional orientation of the document but also quantify it. However GEI selectively samples words instead of using all the words in a document for feature extraction. In order to achieve this GEI takes into account words that have emotional intensity above or equal to the threshold δ (0.25 here) from $ELex$. The transformation results in a modified $ELex$ as follows:

$$GEI_{ELex} = \begin{array}{cc} & \begin{array}{ccccc} e_1 & e_2 & e_3 & e_4 & e_5 \end{array} \\ \begin{array}{c} w_1 \\ w_2 \\ w_3 \\ w_4 \\ w_5 \\ w_6 \end{array} & \begin{pmatrix} 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.4422 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.9255 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.4564 & 0.0 \\ 0.4288 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.5114 \end{pmatrix} \end{array} \tag{5.22}$$

Thereafter the word-document frequencies captured in the WDF matrix to combined with GEI_{ELex} to generate the GEI feature vectors for documents in D. For example the GEI feature vector for document d_2, i.e. d_{2GEI} is obtained by applying the transpose of its corresponding frequency vector from WDF, i.e. <1, 2, 2, 0, 3, 1> (C) as a multiplication filter across the columns of the matrix GEI_{ELex}, followed by the sum of resultant vectors. More visually the scalar multiplication of C^T across columns of GEI_{ELex} and the summation of the resultant vectors is:

$$d_{2\,GEI} = \left(\begin{array}{cccccc} & e_1 & e_2 & e_3 & e_4 & e_5 \\ w_1 & 0 & 0 & 0 & 0 & 0 \\ & & & + & & \\ w_2 & 0 & 0.8844 & 0 & 0 & 0 \\ & & & + & & \\ w_3 & 0 & 0 & 1.8510 & 0 & 0 \\ & & & + & & \\ w_4 & 0 & 0 & 0 & 0 & 0 \\ & & & + & & \\ w_5 & 1.2866 & 0 & 0 & 0 & 0 \\ & & & + & & \\ w_6 & 0 & 0 & 0 & 0 & 0.5114 \end{array} \right) \tag{5.23}$$

(5.24)

$$= (1.2866, 0.8844, 1.8510, 0, 0.5114) \tag{5.25}$$

5.5 Chapter Summary

In this chapter, we first motivated the need for novel feature extraction for emotion classification. Second, we formalized different lexicon based features that utilize the knowledge of a DSEL and also visually presented the lexicon-based feature extraction process using matrix-vector notations. Finally we present the design of hybrid features for emotion classification. Evaluation of proposed lexicon based features, hybrid features is presented in Chapter 7 through emotion classification experiments on benchmark datasets. We evaluate the quality of the proposed lexicon based features extracted using the proposed UMM DSEL in comparison with those extracted using the knowledge of GPELs and DSELs generated using state-of-the-art methods such as point-wise mutual information (PMI) and latent dirichlet allocation (LDA).

Chapter 6

Emotion-corpus guided Lexicons for Twitter Sentiment Analysis

In this chapter we first establish the relationship between emotions and sentiments. Thereafter we propose two different methods, which utilize a corpus of emotion labelled documents for extracting a domain specific sentiment lexicon. We investigate the relationship between emotions and sentiment in the context of social media, where there is emotion-rich content in abundance. The proposed methods, for sentiment extraction adopt an emotion corpus of tweets, to learn Twitter sentiment lexicons. Further such lexicons are applied for Twitter sentiment analysis.

6.1 Relationship between Emotions and Sentiments

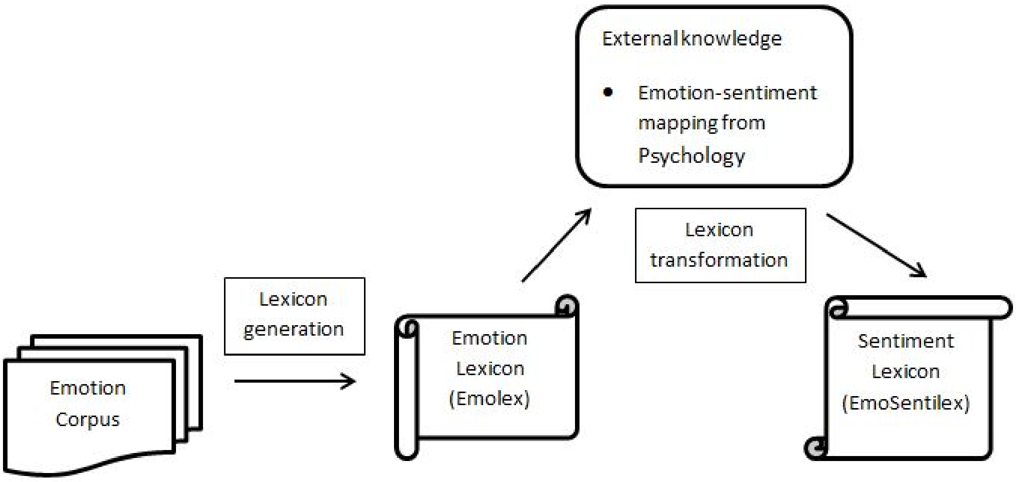
Sentiment analysis concerns the computational study of natural language text (e.g. words, sentences and documents) in order to identify and effectively quantify its polarity (i.e positive or negative) [7]. Sentiment lexicons are the most popular resources used for sentiment analysis, since they capture the polarity of a large collection of words. These lexicons are either hand-crafted (e.g. opinion lexicon [9], General Inquirer [10] and MPQA subjectivity lexicon [11]) or generated (e.g. SentiWordNet [12] and SenticNet [13]) using linguistic resources such as WordNet [14] and ConceptNet [15]. However, on social media (e.g. Twitter), text contains special symbols resulting in non-standard spellings, punctuations and capitalization; sequence of repeating characters and emoticons for which the aforementioned lexicons have limited or no coverage.

As a result domain-specific sentiment lexicons were developed to capture the informal and creative expressions used on social media to convey sentiment [16, 17]. The extraction of such lexicons is possible with limited effort, due to the abundance of weakly-labelled sentiment data on social media, obtained using emoticons [18, 19]. However, sentiment on social media is not limited to conveying positivity and negativity. Socio-linguistics suggest that on social media, people express a wide range of emotions such as *anger, fear, joy, sadness* etc [69]. Following the trends in lexicon based sentiment analysis, research in the textual emotion detection also developed lexicons that can not only capture the emotional orientation of words [5, 70], but also quantify their emotional intensity [43, 46].

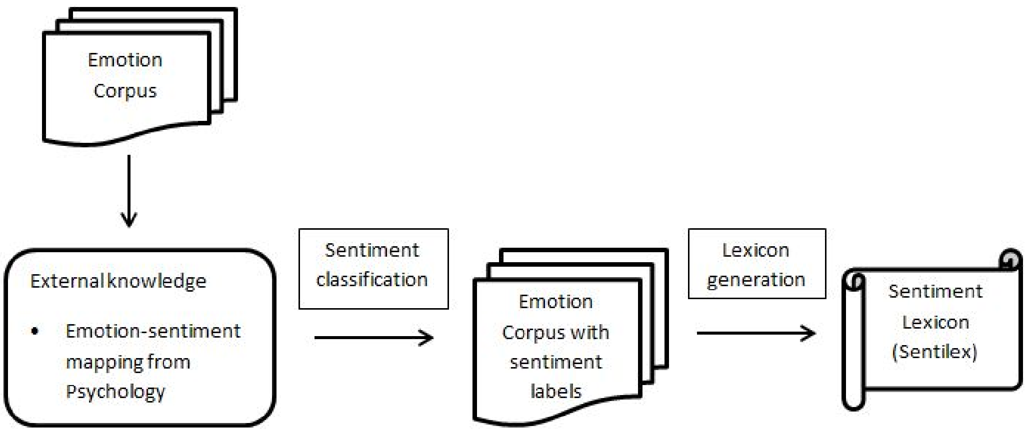
Though research in psychology defines sentiment and emotion differently [34], it also provides a relationship between them [31]. Further research in emotion classification [28, 63] demonstrated the usefulness of sentiment features extracted using a lexicon for document representation. Similarly emoticons used as features to represent documents improved sentiment classification [16, 19]. However, the exploration of emotion knowledge for sentiment analysis is limited to emoticons [19, 35, 36], leaving a host of creative expressions such as emotional hashtags (e.g. #loveisbliss), elongated words (e.g. haaaappyy!!!) and their concatenated variants unexplored. An emotion-corpus crawled on Twitter using seed words for different emotions as in [28, 48] can potentially serve as a knowledge resource for sentiment analysis. Adopting such corpora for sentiment analysis, e.g. sentiment lexicon extraction is particularly interesting, given the challenges involved in developing effective models which can cope with the lexical variations on social media.

6.2 Emotion-Aware Models for Sentiment Analysis

In this section we formulate two different methods which utilize a corpus of emotion-labelled documents for sentiment analysis of text. The first method learns an emotion lexicon and further transforms it into a sentiment lexicon using the emotion-sentiment mapping (refer section 2.2.2 in chapter 2) proposed in Psychology. The second method on the other hand learns the sentiment



(A) Emotion Corpus-EmoSentilex



(B) Emotion Corpus-Sentilex

FIGURE 6.1: Emotion-Aware Models for Sentiment Analysis

labels for the documents in the emotion corpus using the emotion-sentiment mapping, followed by a sentiment lexicon extraction. The two proposed methods are illustrated visually in Figures 6.1a and 6.1b.

6.2.1 *Emotion Corpus-EmoSentilex*

A simple way to utilize a corpus of emotion-labelled documents, X_E for sentiment analysis is to first learn an emotion lexicon, and further transform it into a sentiment lexicon. An emotion lexicon $Emolex$ in our case is a $|V| \times (k+1)$ matrix, where $Emolex(i,j)$ is the emotional valence of the i^{th} word in vocabulary V to the j^{th} emotion in E (set of emotions) and $Emolex(i,k+1)$

corresponds to its neutral valence (refer chapter 4). Further using the emotion-sentiment mapping proposed in psychology we transform the emotion lexicon $Emolex$ into a sentiment lexicon $EmoSentilex$, which is a $|V| \times 1$ matrix as follows:

$$EmoSentilex(i) = Log\left(\frac{\sum_{m \in E^+} Emolex(i,m)}{\sum_{n \in E^-} Emolex(i,n)}\right) \tag{6.1}$$

where $E^+ \subset E$ and $E^- \subset E$ are the set of positive and negative emotions according to the emotion-sentiment mapping. In this research we consider emotions *anger, sadness and fear* as negative emotions, whereas emotions *joy, surprise and love* as positive. Note that the log scoring assigns a positive value for words having stronger associations with emotions such as *joy, surprise and love* and negative values for words having stronger associations with emotions such as *anger, sadness and fear*. Therefore we expect that sentiment knowledge for words is implicitly captured in an emotion lexicon, which can be easily extracted using this simple transformation.

Using the above method, any automatically generated emotion lexicon can be converted into a sentiment lexicon. This is very useful on Twitter, since data (tweets) corresponding to the lexicons is not always available. Further it can also avoid the additional overheads involved in re-crawling the original data using the Twitter API. However, the above method does not model the document-sentiment relationships to learn the lexicon, which is important to quantify word-sentiment associations. Therefore we introduce an alternate method which overcomes this limitation while utilizing an emotion corpus for sentiment lexicon generation.

6.2.2 *Emotion Corpus-Sentilex*

An alternate way to utilize the emotion corpus, X_E for sentiment analysis is to transform it into a sentiment corpus, X_S by learning the sentiment label for each document $d \in X_E$. This is done by using the emotion-sentiment mapping as follows:

$$Sentiment(d) = \begin{cases} positive & \text{if emotion(d)} \in E^+ \\ negative & \text{if emotion(d)} \in E^- \end{cases} \tag{6.2}$$

After the sentiment label for a document is obtained, we model each document in the corpus X_S to be a mixture of sentiment bearing words and neutral (background) words. This assumption is reasonable, since an emotion-rich corpus also conveys sentiment but in a finer level of positive and negative concepts, such as *joy, surprise, anger, sadness* etc. Therefore we propose a generative model which assumes a mixture of two unigram language models to account for such word mixtures in documents. More formally our generative model is as follows to describe the generation of documents connoting sentiment Pos, D_{Pos} as follows (similarly for negative documents D_{Neg}):

$$P(D_{Pos}, Z|\theta_{Pos}) = \prod_{i=1}^{|D_{Pos}|} \prod_{w \in d_i} [(1 - Z_w)\lambda_{Pos} P(w|\theta_{Pos}) + (Z_w)(1 - \lambda_{Pos}) P(w|N)]^{c(w,d_i)} \tag{6.3}$$

where θ_{Pos} is the sentiment language model and N is the background language model. λ_{Pos} is the mixture parameter and Z_w is a binary hidden variable which indicates the language model that generated the word w.

The estimation of parameters θ_{Pos} and Z is done using expectation maximization (EM), which iteratively maximizes the complete data (D_{Pos}, Z) by alternating between E-step and M-step. The E and M steps in our case are as follows:

E-step:

$$P(Z_w = 0|D_{Pos}, \theta_{Pos}^{(n)}) = \frac{\lambda_{Pos} P(w|\theta_{Pos}^{(n)})}{\lambda_{Pos} P(w|\theta_{Pos}^{(n)}) + (1 - \lambda_{Pos}) P(w|N)} \tag{6.4}$$

M-step:

$$P(w|\theta_{\theta_{Pos}}^{(n+1)}) = \frac{\sum_{i=1}^{|D_{Pos}|} P(Z_w = 0|D_{Pos}, \theta_{Pos}^{(n)}) c(w, d_i)}{\sum_{w \in V} \sum_{i=1}^{|D_{Pos}|} P(Z_w = 0|D_{Pos}, \theta_{Pos}^{(n)}) c(w, d_i)} \tag{6.5}$$

where n indicates the EM iteration number. The EM iterations are terminated when an optimal estimate for the sentiment language model θ_{Pos} is obtained. Similarly, EM is used to estimate the parameters of the mixture model corresponding to negative sentiment (Neg). Thereafter, the sentiment lexicon $Sentilex$ is learnt by using the two sentiment language models ($\theta_{Pos}^{(n)}$, $\theta_{Neg}^{(n)}$) and the background model N as follows:

$$Sentilex(w_i, \theta_{Pos}) = \frac{P(w_i|\theta_{Pos}^{(n)})}{P(w_i|\theta_{Pos}^{(n)}) + P(w_i|\theta_{Neg}^{(n)}) + P(w_i|N)} \tag{6.6}$$

$$Sentilex(w_i, \theta_{Neg}) = \frac{P(w_i|\theta^{(n)}_{Neg})}{P(w_i|\theta^{(n)}_{Pos}) + P(w_i|\theta^{(n)}_{Neg}) + P(w_i|N)} \tag{6.7}$$

$$Sentilex(w_i, N) = \frac{P(w_i|N)}{P(w_i|\theta^{(n)}_{Pos}) + P(w_i|\theta^{(n)}_{Neg}) + P(w_i|N)} \tag{6.8}$$

where $Sentilex$ is a $|V|{\times}3$ matrix, and $Sentilex(i, Pos)$, $Sentilex(i, Neg)$ and $Sentilex(i, N)$ are the positive, negative and neutral valences corresponding to the i^{th} word in vocabulary V. Observe that unlike the method which learns $EmoSentilex$, by aggregating word-level emotion scores into sentiment scores, this method learns the sentiment-class knowledge corresponding to the documents, before learning a word-sentiment lexicon. We expect this additional layer of supervision to improve performance in sentiment analysis. Further details about our proposed lexicon generation method can be found in Chapter 4

6.3 Sentiment Lexicon Generation: A Walk through Example

In this section, we illustrate the various steps involved in the sentiment lexicon generation using the proposed method, with the help of sample Twitter data. The data used to train the lexicon for the two sentiment classes, *positivity and negativity* is shown in Tables 6.1 and 6.2.

6.3.1 Initial language model generation

The initial language models corresponding to the sentiment classes *positivity* and *negativity* and the background language model are generated according to equations 4.20 and 4.21 in section 4.3.3. At the end of this step, the initial language models $\theta^{(0)}_{Pos}$, $\theta^{(0)}_{Neg}$ and the background model N are generated. A sample of these language models on the toy data set (*positive*, *negative* documents) is shown in Table 6.3.

6.3.2 Parameter Estimation

In this section, we illustrate the estimation process for the parameters λ_{Pos}, λ_{Neg}, θ_{Pos}, θ_{Neg} and Z. In order to estimate the optimal values for λ_{Pos} and λ_{Neg}, we observe the likelihood of the unseen data (development data set) shown in Tables 6.1 and 6.2 according to the formulations

Positivity training set	Positivity validation set
1.i love going to bed with a smile on my face! :)	1. today was a very good day
2.i had a great sunday	2.had great church services today!
3.what a great weekend!	3. i gotta start packing soon :)
4.gonna have a good day :)	4. falling in love with you was the best choice i have made in a long time.
5.finally i feel like everything is turning out good for once :)	5. perfect way to celebrate!xxxx
6.love everyone that is apart of my life!!	
7.can't wait to see my knucklehead in detroit in a few months!!!! lol	
8.my life is so perfect right now	
9.i love black friday shopping. #more	

TABLE 6.1: Positivity documents

Negativity training set	Negativity validation set
1.can't sleep :-(	1. wtf you got me fuccccckkkeeeddd up !
2.i have neva met someone as immature as this creature	2. hate trying to wrap presents up that have an awkward shape.
3.too many dumbasses at the gym	3. i don't understand how someone can be so immature
4.ughhhh people who lick stuff off their fingers in resturants bother me.	4. hate it when plans fail.... :(
5.that's totally fucked up	
6.i'm really hating this :(#confused	
7.anyways wtf is wrong with my fone wth this squeeky ass noise	
8.i hate freakin losing knowing that we should have won..	

TABLE 6.2: Negativity documents

Words	$\theta^{(0)}_{Pos}$	$\theta^{(0)}_{Neg}$	N
love	0.0227	0.0060	0.0146
smile	0.0113	0.0060	0.0087
⋮	⋮	⋮	⋮
that	0.0113	0.0120	0.0116
hate	0.0056	0.0120	0.0087
freakin	0.0042	0.0197	0.0065
⋮	⋮	⋮	⋮

TABLE 6.3: Initial language models

in equations 4.4 and 4.5. In this research, we experimented with 11 different values [0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0] of λ and selected the one, which maximizes the log-likelihood of unseen data. As mentioned before λ is inversely proportional to the noise in the documents. Therefore, if λ for a sentiment class is closer to 1, its documents are more sentiment-rich.

Tables 6.4 and 6.5 capture the log likelihood values for different values of λ across three EM iterations. It is evident from these tables that the initial language models $\theta^{(0)}_{Pos}$ and $\theta^{(0)}_{Neg}$ best generate the unseen data for λ values 0.8 and 0.9 respectively. This shows that the assumption of real world data to be a mixture of sentiment-bearing and sentiment-neutral words is valid. In the E-step (refer equation 6.4), the optimum values for λ are substituted to estimate the probability values for the hidden variables Z being zero or not. Thereafter those probability values are used in the M-step (refer equation 4.19) to estimate the language models $\theta^{(1)}_{Pos}$, $\theta^{(1)}_{Neg}$ and similarly $\theta^{(2)}_{Pos}$, $\theta^{(3)}_{Pos}$ and $\theta^{(2)}_{Neg}$, $\theta^{(3)}_{Neg}$. Table 6.6 shows the updated language models after three EM iterations. Since, the proposed method for lexicon generation is applied on a tiny data set, convergence of the EM iterations happened very quickly. However on a real world data set, it is expected to see more iterations before convergence. We illustrate our findings on real world data sets in Chapter 7.

$\lambda_{\theta_{Pos}}$	Likelihood for $\theta^{(0)}_{Pos}$	Likelihood for $\theta^{(1)}_{Pos}$	Likelihood for $\theta^{(2)}_{Pos}$	Likelihood for $\theta^{(3)}_{Pos}$
0.0	-73.1542	-73.1542	-73.1542	-73.1542
0.1	-72.6818	-72.6370	-72.6317	-72.6310
0.2	-72.2341	-72.1496	-72.1394	-72.1381
0.3	-71.8093	-71.6893	-71.6749	-71.6729
0.4	-71.4058	-71.2541	-71.2358	-71.2333
0.5	-71.0220	-70.8422	-70.8204	-70.8174
0.6	-70.6568	-70.4521	-70.4273	-70.4238
0.7	-70.3092	-70.0826	-70.0551	-70.0512
0.8	-69.3644	-69.0890	-69.0553	-69.0505
0.9	-69.4637	-69.4018	-69.3698	-69.3652
1.0	-69.5784	-69.4328	-69.4028	-69.3986

TABLE 6.4: Positivity Log Likelihood Estimates

$\lambda_{\theta_{Neg}}$	Likelihood for $\theta^{(0)}_{Neg}$	Likelihood for $\theta^{(1)}_{Neg}$	Likelihood for $\theta^{(2)}_{Neg}$	Likelihood for $\theta^{(3)}_{Neg}$
0.0	-59.5035	-59.5035	-59.5035	-59.5035
0.1	-59.2884	-59.2688	-59.2660	-59.2655
0.2	-59.0883	59.0521	-59.0470	-59.04614
0.3	-58.9030	-58.8531	-58.8460	-58.8448
0.4	-58.7323	-58.6716	-58.6629	-58.6614
0.5	-58.5760	-58.5074	-58.4976	-58.4959
0.6	-58.4342	-58.3607	-58.3502	-58.3484
0.7	-58.3072	-58.2321	-58.2214	-58.2195
0.8	-58.1954	-58.1221	-58.1118	-58.1099
0.9	-58.0202	-57.9634	-57.9557	-57.9542
1.0	-58.0994	-58.0320	-58.0226	-58.0209

TABLE 6.5: Negativity Log Likelihood Estimates

Words	$\theta^{(1)}_{Pos}$	$\theta^{(1)}_{Neg}$	$\theta^{(2)}_{Pos}$	$\theta^{(2)}_{Neg}$	$\theta^{(3)}_{Pos}$	$\theta^{(3)}_{Neg}$	N
love	0.0235	0.0052	0.0236	0.0051	0.0236	0.0050	0.0146
smile	0.0116	0.0057	0.0116	0.0057	0.0116	0.0057	0.0087
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
that	0.0113	0.0120	0.0112	0.0122	0.0112	0.0122	0.0116
hate	0.0053	0.0123	0.0053	0.0124	0.0053	0.0124	0.0087
freakin	0.0043	0.0196	0.0042	0.0197	0.0041	0.0198	0.0065
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

TABLE 6.6: Sentiment language models over three EM iterations

6.3.3 Lexicon Generation

Lexicon generation is done using the optimal language models for sentiments *positivity, negativity* i.e. $\theta^{(3)}_{Pos}$ and $\theta^{(2)}_{Neg}$ and the background (neutral)language model N as shown in equations 4.22 and 4.23. Table 6.7 shows the sentiment lexicon obtained by normalizing the language models, $\theta^{(3)}_{Pos}$, $\theta^{(3)}_{Neg}$ and N. Observe that sentiment bearing words such as *love* and *smile* are assigned high scores with sentiment *positivity*, where as words such as *hate* and *freakin* are assigned high scores with sentiment *negativity*. Further sentiment-neutral word such as *that* is penalized by the background language model, since it is neither associated with *positivity* nor *negativity*. The proposed mixture model is able to capture such word-mixtures present in the data and quantify their sentiment accordingly. We expect such ability to model sentiment at word-level is useful for performance in different sentiment analysis tasks, which are discussed later (refer Chapter 7).

6.4 Chapter Summary

In this chapter we first highlighted the relationship between sentiments and emotions. Thereafter we introduced two different methods, which utilize an emotion-corpus of tweets and an emotion-sentiment mapping from psychology to learn word-sentiment lexicons for sentiment analysis of tweets. The proposed methods are generic and can be applied to any domain, that is sentiment-rich as well as emotion-rich. The evaluation of the learnt word-sentiment lexicons is presented

Words	Positivity	Negativity	Neutral
love	0.5453	0.1172	0.3374
smile	0.4456	0.2187	0.3355
⋮	⋮	⋮	⋮
that	0.3226	0.3442	0.3331
hate	0.2013	0.4679	0.3307
freakin	0.1348	0.6513	0.2138
⋮	⋮	⋮	⋮

TABLE 6.7: UMM Emotion Lexicon

in chapter 7 through different sentiment analysis tasks: *sentiment classification and sentiment intensity prediction* on benchmark Twitter data sets. We evaluate the quality of the proposed word-sentiment lexicons in comparison with existing lexicons for Twitter sentiment analysis.

Chapter 7

Evaluations

In this chapter we present the evaluations concerning the different algorithms proposed in chapters 4, 5 and 6 for emotion detection from text. Firstly we formally describe the different evaluation tasks concerning emotion detection and sentiment analysis. Secondly we present the performance evaluation of different emotion lexicons including the proposed one at detecting emotion at word and phrase level. Thereafter we show the performance analysis of different feature extraction techniques which rely on the knowledge of the proposed emotion lexicon in comparison with the other standard features used for emotion classification. Finally we present the results for the emotion-sentiment interplay tasks, studied through utilizing the knowledge of an emotion-corpus for sentiment lexicon extraction in order to perform sentiment analysis of tweets.

7.1 Evaluation Tasks

In this section we formally present the different evaluation tasks used in this research to assess the performance of the baseline methods and proposed methods for emotion detection and sentiment analysis.

7.1.1 Word-Emotion Classification

The most obvious way to evaluate a word-emotion lexicon is to classify a collection of target words hand labelled with emotions. More formally given an arbitrary word w the task is to predict an emotion label $e \in E$ for w using the word-emotion lexicon, where E is a predefined set of emotions.

7.1.2 Document-Emotion Ranking

Subjective textual content usually captures one or more emotions. Therefore words in emotion corpora have associations with multiple emotions with varying magnitude. More formally given a sentence s, expressing emotions $(e_1, \ldots, e_m)$ in decreasing order of magnitude, the task is to predict the order of emotions for s using a lexicon. This task measures not just the ability of a lexicon in predicting the dominant emotion in s, but also the residual other emotions. For any given phrase or a sentence s, an emotion ranking could be formed by an ordered list of emotions expressed by s, $(e_1, \ldots, e_m) \mid$ for $i, j \in (1, m)$, if $i < j$, then $s[e_i] > s[e_j]$, where $s[e]$ is calculated using the lexicon as follows:

$$s[e] = \sum_{w \in s} Lex(w, e) \times count(w, s) \tag{7.1}$$

where $count(w, s)$ denotes the number of times w appears in s.

7.1.3 Document-Emotion Classification

Given a collection of documents, the objective is to classify them into predefined emotion classes such as *anger, fear, joy, sadness*. Typically, machine learning approaches are observed to give the best performance in emotion text classification. Therefore in this research we define emotion classification as a machine learning task. Formally, given a document d, a machine learning approach involves an intermediary step to learn a representation for the documents, also known as a feature vector. Let d_{vec} be the feature vector corresponding to d. d_{vec} could be learnt using any of the methods discussed in Chapter 5. The feature vectors for the training documents d_{train} are used to learn a classifier C. Finally the emotion class of an unseen document d_{test} is determined heuristically. For example, in the case of support vector machines (SVM), the sign of the product $d_{test}.W + b$, where W and b are the parameters of an SVM.

7.1.4 Sentiment Intensity Prediction

Given a collection of words/phrases extracted from sentiment bearing tweets, the objective is to predict a sentiment intensity score for each word/phrase and arrange them in decreasing order of intensity. The predictions are validated against a ranking given by humans. Formally, given a phrase P, the sentiment intensity score for the phrase is calculated as follows:

$$SentimentIntensity(P) = \sum_{w \in P} Log\left(\frac{Lex(w,+)}{Lex(w,-)}\right) \times count(w,P) \tag{7.2}$$

where w is a word in the phrase P, $count(w,P)$ is the number of times w appears in P. $Lex(w,+), Lex(w,-)$ are the positive and negative valences for the word w in a lexicon.

7.1.5 Sentiment Classification

Given a collection of documents (tweets), the objective is to classify them into positive and negative classes. The predictions are validated against human judgements. Formally, given a document d, the sentiment class is predicted using a lexicon as follows:

$$d[+] = \sum_{w \in d} Lex(w,+) \times count(w,d) \tag{7.3}$$

where $d[+]$ is the positive intensity of d. Similarly $d[-]$ indicates the negative intensity of d. Finally the sentiment class of d is determined as follows:

$$Sentiment(d) = \begin{cases} positive & \text{if } d[+] > d[-] \\ negative & \text{if } d[-] > d[+] \end{cases} \tag{7.4}$$

7.2 Evaluation Results

In this section we present the empirical results concerning all the evaluation tasks detailed earlier. Our evaluation typically is a comparative analysis of the performance of the proposed methods and the existing methods in literature (baselines). In each evaluation tasks, we conduct a pair-wise t-test between the proposed method and all the baselines and report statistical significance of the performance improvements. Significance is reported using a paired one-tailed t-test using 95% confidence (i.e. with p value $<= 0.05$). Throughout the evaluation, the best performing method for an evaluation task is highlighted in bold. Further we also explain in detail about the performance improvements of a method by an in-depth analysis of its characteristics. Before proceeding with the evaluation results we outline below the different methods that are compared for performance on each emotion detection task. The details are as follows:

1. *Word-Emotion Classification*: In this task we comparatively evaluate the performance of different GPELs such as ESN, NRC and WNA and DSELs such as WED, sLDA, PMI and UMM in classifying words into predefined emotion classes. The evaluation is carried out on the blogs data set presented in section 3.6 of chapter 3.

2. *Document-Emotion Ranking*: In this task we comparatively assess the performance of the DSELs WED, sLDA, PMI and UMM in predicting the order of emotions associated with each document. The evaluation is carried out on the Twitter events data set presented in section 3.6 of Chapter 3

3. *Document-Emotion Classification*: In this task we comparatively evaluate the quality of different document representations proposed in literature for emotion classification, emotion lexicon based representations extracted using the knowledge of baseline lexicons such as PMI, LDA and the proposed UMM based lexicon. We evaluate the different representations for individual class (emotion) performance and also overall performance. Based on this evaluation we construct hybrid representations by combining the best performing baseline features and the best performing lexicon based features. We combined the best performing features (i.e. baseline, lexicon based features) to construct hybrid features expecting further performance improvements. This evaluation is carried out on the blogs, SemEval-07, Twitter and ISEAR data sets presented in section 3.6 of Chapter 3.

4. *Sentiment-Intensity Prediction*: In this task we comparatively evaluate the quality of standard sentiment lexicons such as SentiwordNet, SenticNet, S140 lexicon [16], NRCHashtag lexicon [16], UMM based sentiment lexicons learnt on S140 Twitter sentiment corpus (refer chapter 6)and emotion-corpus based sentiment lexicons such as $EmoSentilex$ and $Sentilex$ proposed in chapter 6. The evaluation is carried out on SemEval-2015 data set presented in section 3.6 of Chapter 3.

5. *Sentiment Classification*: In this task we comparatively evaluate the same lexicons as in the case of sentiment intensity prediction. The evaluation is carried out on S140 and SemEval-2013 data sets presented in section 3.6 of Chapter 3.

6. *Perplexity Analysis*: In this task we assess the quality of the language models (topics) learnt by the generative lexicons such as sLDA and UMM. In this task we use the blogs, SemEval-07, Twitter and ISEAR data sets presented in section 3.6 of Chapter 3.

7.2.1 Parameter Tuning

In this section we illustrate the estimation of parameter λ corresponding to the mixture model of emotion e_k. Let $D^{dev}_{e_k}$ be the development data corresponding to emotion e_k. The best λ_{e_k} is the one that maximizes the log-likelihood of $D^{dev}_{e_k}$ as follows:

$$\hat{\lambda}_{e_k} = \underset{\lambda_{e_k}}{\text{argmax}} logL(\theta_{e_k}) \tag{7.5}$$

$$\begin{aligned} logL(\theta_{e_k}) &= logP(D^{dev}_{e_k}|\theta_{e_k}) \\ &= \sum_{i=1}^{|D^{dev}_{e_k}|} \sum_{w \in d_i} c(w, d_i) log[\lambda_{e_k} P(w|\theta_{e_k}) \\ &+ (1 - \lambda_{e_k}) P(w|N)] \end{aligned} \tag{7.6}$$

where θ_{e_k} is learnt on the training data $D^{train}_{e_k}$. We experimented with different values[1] of λ and selected the one, which maximizes the log-likelihood of $D^{dev}_{e_k}$. As mentioned before λ is

[1] [0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0]

Emotion	Max λ	#EM iterations
Surprise	1.0	3
Anger	0.9	3
Fear	0.9	3
Joy	0.9	3
Sadness	0.9	3
Disgust	0.8	3

TABLE 7.1: Parameter Tuning on News Headlines (SemEval-07)

Emotion	Max λ	EM iterations
Anger	0.9	3
Joy	0.9	3
Surprise	0.9	3
Sadness	0.8	3
Fear	0.7	3

TABLE 7.2: Parameter Tuning on Blogs

Emotion	Max λ	#EM iterations
Anger	0.9	5
Joy	0.9	5
Love	0.9	5
Sadness	0.8	5
Surprise	0.3	7
Fear	0.1	7

TABLE 7.3: Parameter Tuning on Tweets

inversely proportional to the noise in the documents. Therefore, if λ for an emotion is closer to 1, means that its documents are highly emotion-rich. Tables 7.1 to 7.4 show the optimal λ obtained for each emotion on news, blogs, tweets and incident reports respectively. It is evident from the analysis that for most of the emotions, the optimum value for λ is less than 1, thus indicating the

Emotion	Max λ	#EM iterations
Anger	0.9	3
Disgust	0.9	3
Fear	0.8	4
Guilt	0.7	5
Joy	0.8	4
Sadness	0.8	4
Shame	0.8	3

TABLE 7.4: Parameter Tuning on Incident Reports (ISEAR)

noisy nature of real-world emotion data and the need for a mixture model which models both the emotionality and neutrality of documents at the word-level. In general we found emotions such as *anger, joy and sadness* have less noise compared to emotions such as *disgust, fear and surprise*. Further the tables also show the number of EM iterations taken to find the optimum values for the parameters (θ and Z) of the mixture model defining an emotion. We observed that on Twitter, which has loosely-labelled emotion data, EM iterations converge late, in contrast to news, blogs and incident reports, which are manually-labelled with emotions. This is expected because manually assigned class labels, provide more accurate initial values for EM, thereby leading to faster convergence. In the following section we compare the quality of the language models (topics) obtained using UMM and sLDA algorithms respectively using a standard metric known as perplexity. This is useful to understand the quality of the respective lexicons that are generated using the language models.

7.2.2 Perplexity Analysis

In this section we present the results for the perplexity analysis. Perplexity is the per-word average of the probability with which a language model generates the test data, where the average taken is over the number of words in the test data.

Since sLDA and UMM are generative models, they capture the associations between words and emotions in the form of probability distributions, which are further transformed to obtain an

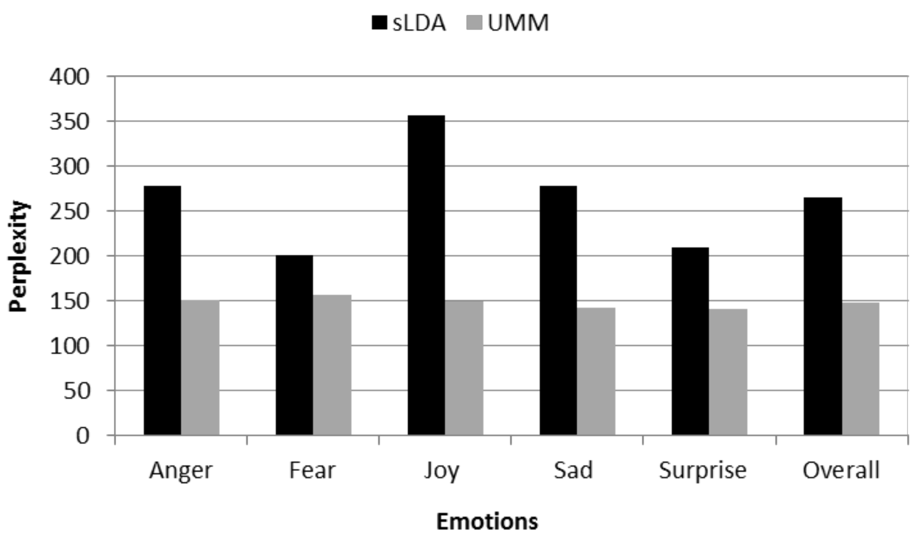


FIGURE 7.1: Perplexity scores of emotion topics on Blogs

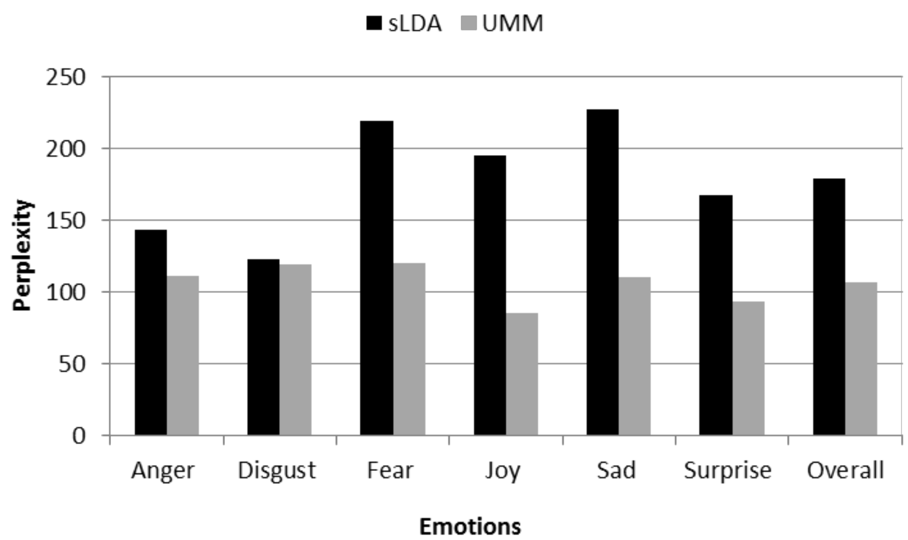


FIGURE 7.2: Perplexity scores of emotion topics on News (SemEval-07)

emotion lexicon. Therefore assessing the quality of the language models gives deeper insights about the effectiveness of the resulting lexicons. In our evaluation we compare the language models (topics) generated by UMM and sLDA for each emotion. Perplexity scores for sLDA and UMM[2] based emotion language models (topics) on blogs, news, tweets and incident reports are shown in Figures 7.1, 7.2, 7.3 and 7.4 respectively. UMM emotion topics were found to have significantly lower perplexity than those of sLDA on all the four data sets, suggesting the superiority of UMM over sLDA in characterising emotional documents. The ability of UMM

[2]Perplexity analysis is done on the language models from the final EM iteration.

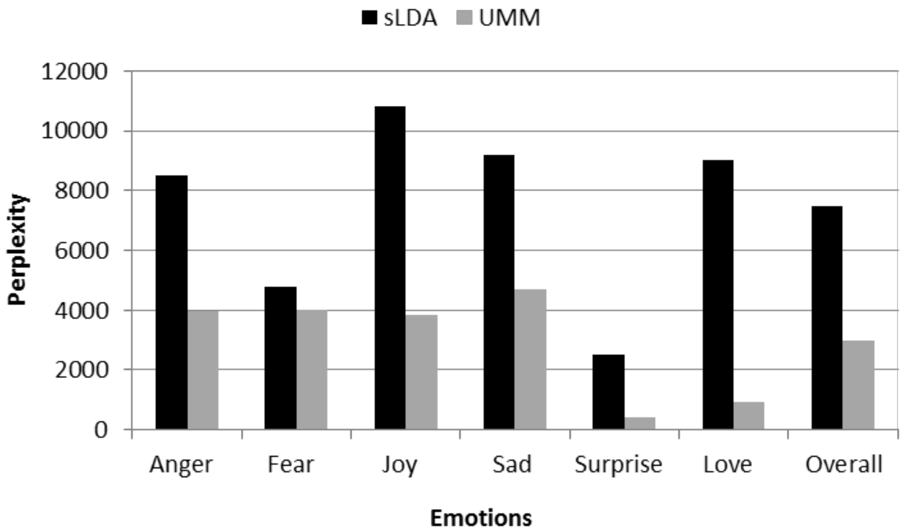


FIGURE 7.3: Perplexity scores of emotion topics on Twitter

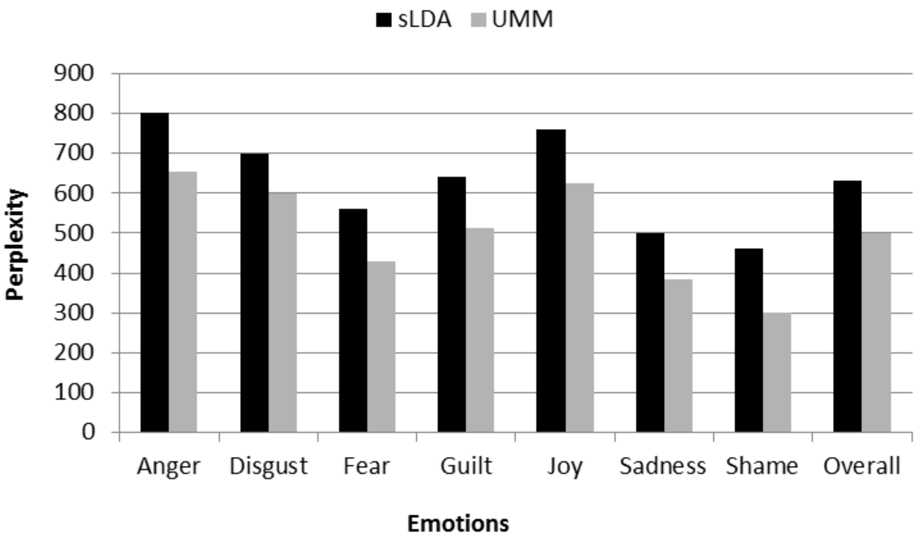


FIGURE 7.4: Perplexity scores of emotion topics on incident reports (ISEAR)

to iteratively refine the emotion language models in order to maximize the likelihood of data resulted in performance improvements over sLDA. In order to get a deeper understanding of the performance of the different lexicons, we analysed the most expressive words for each emotion identified by the different lexicon generation methods. We present the details of this analysis in the following section.

7.2.3 Emotion word clouds for Lexicons

In this section we analyse the word-emotion associations learnt by the different lexicons, WED, PMI, sLDA and UMM lexicon from the training data corresponding to the blog data set. This data set is particularly interesting, given its small size and the skewed emotion-class distribution (refer chapter 3). Further we expect the word-level analysis to reveal interesting trends that could effect the knowledge (e.g. lexicon based document representations) extracted from the word-emotion lexicons. Figures 7.5 to 7.24 show the most expressive words for emotions *anger, fear, joy, sadness and surprise* identified by WED, sLDA, PMI and UMM lexicons. It is evident from the figures that unlike the GPELs, all these lexicons capture the domain-specific vocabulary that is expressed informally. This is very important for effective emotion detection in a domain.

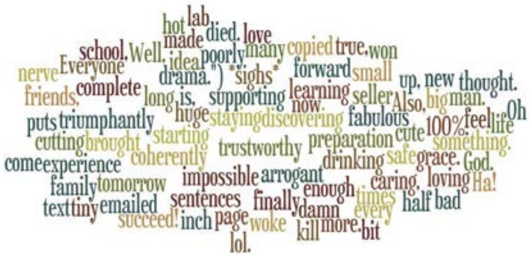


FIGURE 7.5: Top anger words by WED lexicon

FIGURE 7.6: Top anger words by sLDA lexicon

FIGURE 7.7: Top anger words by PMI lexicon

FIGURE 7.8: Top anger words by UMM lexicon

The word clouds presented in the figures are the top 100 words for each emotion, after removing the common words in English language. We observed that WED lexicon is biased towards the majority class (*joy* here) in the corpus in learning the word-emotion associations. For example it identified words connoting *joy* such as *succeed!* and *Ha!* as top *anger* words, similarly for other emotions. This is due to the fact that WED lexicon is designed for emotion rated documents and it is less effective in capturing word-emotion associations on a corpus that have discrete emotion labels. On the other hand sLDA lexicon, because of the assumption of its underlying generative model that documents are a mixture of multiple topics (emotions) learnt better word-emotion associations compared to WED lexicon. However sLDA lexicon was not able to discriminate effectively between words that strongly convey a particular emotion and those that are weakly associated with an emotion. For example words such as *scared, worried and nervous* are not well distinguished from other words for emotion *fear* and similarly for other emotions. As a result it was observed in the word clouds for the sLDA lexicon that top words for each emotion have similar size. This is not desirable since the word-emotion association scores form an important knowledge resource for learning document representations for emotion classification. Therefore we expect the representations derived from sLDA to be limitedly effective for emotion detection (e.g. emotion classification).

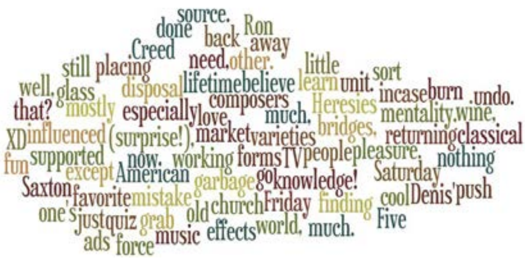


FIGURE 7.9: Top fear words by WED lexicon

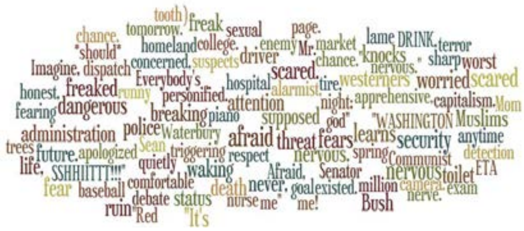


FIGURE 7.10: Top fear words by sLDA lexicon

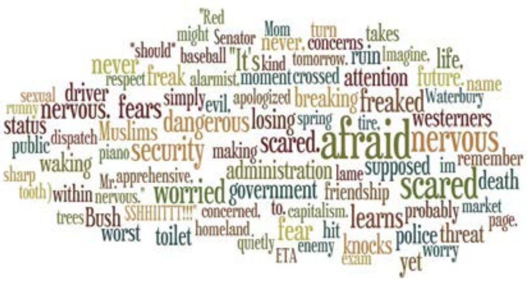


FIGURE 7.11: Top fear words by PMI lexicon

FIGURE 7.12: Top fear words by UMM lexicon

It was observed that PMI and UMM lexicons discriminate between strong and weak words for each emotion effectively. This is very promising, since the lexicon based feature extraction methods will be able to use this knowledge to discriminate between documents that lie close to the class boundaries in emotion text classification. Though PMI and UMM lexicons were

FIGURE 7.13: Top joy words by WED lexicon

FIGURE 7.14: Top joy words by sLDA lexicon

FIGURE 7.15: Top joy words by PMI lexicon

FIGURE 7.16: Top joy words by UMM lexicon

performing closely in identifying top terms for each emotion, we observed that PMI is unable to capture words that occur rarely, but convey emotions. It is very common in domains such as social media to find syntactical variants for words to express emotions and having an index for such is important to have performance gains. Hence we expect the PMI lexicon to not effectively represent documents that contain rare emotional words, which as a result will impact performance in the emotion detection tasks. However UMM is observed to capture words that are emotion-relevant but are rare. For example words such as *:)* and *fun!* for the emotion *joy*, *shit* and *hard* for emotion anger, *used* for emotion sad and *weekend* for emotion *surprise*. We observed similar trends as mentioned above for the rest of the lexicon vocabulary. We expect this word-level analysis to help infer useful insights about the performance gains of the proposed lexicon over the baselines in different emotion detection tasks. In the following section we analyse the performance of the different lexicons at detecting associations between words and emotions.

7.2.4 Word-Emotion Classification Results

Word classification results on Blog data appear in Table 7.5. Here the results are the average overall F-scores obtained over 5 folds. It is evident from the results that UMM lexicon significantly outperformed GPELs (WNA,NRC and ESN) by 23%, 13% and 24%, PMI, sLDA

FIGURE 7.17: Top sad words by WED lexicon

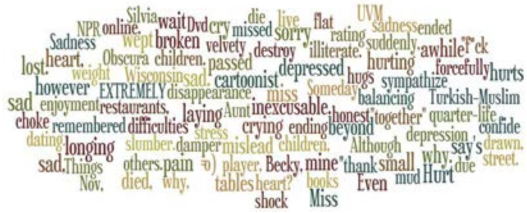


FIGURE 7.18: Top sad words by sLDA lexicon

FIGURE 7.19: Top sad words by PMI lexicon

FIGURE 7.20: Top sad words by UMM lexicon

FIGURE 7.21: Top surprise words by WED lexicon

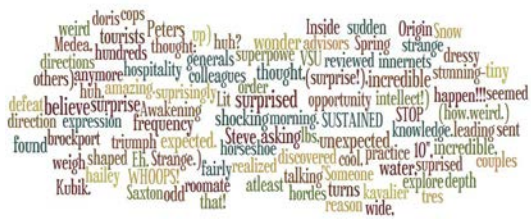


FIGURE 7.22: Top surprise words by sLDA lexicon

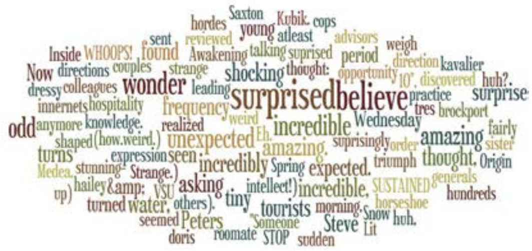


FIGURE 7.23: Top surprise words by PMI lexicon

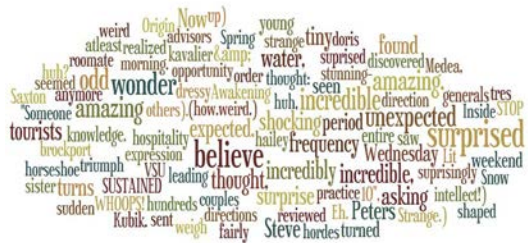


FIGURE 7.24: Top surprise words by UMM lexicon

and WED based domain specific lexicons by 10%, 14% and 28% respectively. This evaluation clearly suggests that GPELs in general are inadequate tools for emotion analysis in a domain without adaptation. In particular the performance of WNA, ESN reflect their low coverage of informal emotion vocabulary, which is very common in domains such as social media (e.g. internet blogs). Though NRC lexicon performed the best among GPELs its static nature makes it an inadequate tool for emotion analysis of domains such as social media.

Method	Avg Overall F-score	p-value at p < 0.5
Baseline GPELs		
WNA	29.96	p < 7.43E-15
NRC	39.05	p < 7.43E-15
ESN	28.30	p < 7.43E-15
Baseline DSELs		
PMI	42.12	p < 7.43E-15
WED	24.51	p < 7.43E-15
sLDA	38.72	p < 7.43E-15
Proposed DSEL		
UMM	**52.84**	n/a

TABLE 7.5: Word-Emotion Classification Results on Blogs

On the other hand WED based DSEL performed below GPELs. We believe the tailoring of the WED lexicon generation towards emotion-rated documents made it less effective for a corpus with discrete emotion labels. We expect it to perform better on the news (SemEval-07) corpus, which has emotion ratings for each document (refer section 7.2.5). Also the assumption of sLDA that documents exhibit multiple emotions proved to be less effective for predicting word-level emotion associations. By far PMI performed the best among the baselines, however the ability of UMM to penalize emotionally neutral words resulted in the best performance in predicting emotions at word-level.

7.2.5 Document-Emotion Ranking Results

DSELs generated using PMI, sLDA, WED and UMM are compared on emotion rank prediction applied to news headlines and to events captured by tweets (see Tables 7.6 and 7.7). As expected on the news (SemEval-07) corpus which has document-level emotion ratings, WED performed significantly better than other baselines, because of its ability to leverage numerical ratings on documents for lexicon induction. In contrast on the events corpus WED lexicon learnt on tweets with discrete emotion labels performed the poorest, thus indicating that it is applicable only to specific emotion corpora (i.e. corpora with numerical labels). Comparing the results of sLDA

Method	MAP	p-value (at p<0.05) for MAP	MRR	p-value (at p<0.05) for MRR
Baseline DSELs				
PMI	64.66	p < 6.23E-16	30.53	p < 3.46E-13
WED	78.10	p < 6.23E-16	53.08	p < 3.46E-13
sLDA	67.44	p < 6.23E-16	35.42	p < 3.46E-13
Proposed DSEL				
UMM	**80.33**	n/a	**56.05**	n/a

TABLE 7.6: Document-Emotion Ranking on News Headlines (SemEval-07) data set

and PMI lexicons on both the corpora suggest that sLDA is more effective when documents exhibit multiple emotion characteristics. On the other hand when documents explicitly connote a single emotion, PMI gives better performance, which is consistent with the findings in the literature.

However we found the UMM lexicon performs consistently and significantly better than the baselines across both the corpora, which suggests its corpus-independent nature and also its effectiveness in transferability across similar emotion corpora. This evaluation also evidenced that UMM is not only accurate in predicting the dominant emotion, but also the sub-dominant emotions in a document. Further both the word-emotion classification and document-emotion ranking tasks demonstrated the superiority of UMM lexicon as a direct tool for emotion analysis at word and document level. In the following section we present the evaluation results for document-emotion classification.

7.2.6 Document-Emotion Classification Results

In this section we analyse the emotion classification results obtained using baseline features, lexicon based features and a combination of them (i.e. hybrid features). We first observe the performance of the baseline features and lexicon based features individually. Thereafter we combine the best performing baseline and lexicon based features to obtain the hybrid features and study their performance for improvements in emotion classification tasks.

Method	MAP	p-value (at p<0.05) for MAP	MRR	p-value (at p<0.05) for MRR
Baseline DSELs				
PMI	64.66	p < 2.12E-10	30.53	p < 3.05E-7
WED	78.10	p < 2.12E-10	53.08	p < 3.05E-7
sLDA	67.44	p < 2.12E-10	35.42	p < 3.05E-7
Proposed DSEL				
UMM	**80.33**	n/a	**56.05**	n/a

TABLE 7.7: Document-Emotion Ranking on Events data set

Baseline features	Overall F-Score			
	SemEval-07	Twitter	Blogs	ISEAR
ngrams	35.77	**49.55**	**58.32**	32.19
ngrams+POS	38.63	46.80	57.15	31.90
ngrams+CF	39.17	48.38	57.60	32.07
ngrams+POS+CF	**40.99**	47.19	57.03	**32.21**

TABLE 7.8: Overall performance on different datasets with baseline features

7.2.6.1 Performance of baseline features

Emotion classification experiments using baseline features were done incrementally by beginning with n-grams and adding one feature group (e.g. POS) at a time. Table 7.8 summarizes the results obtained for baseline features on the four benchmark data sets. Overall performance is measured by combining (average) the macro-averaged F-score of all the emotion classes. In general, the combination of n-grams with POS features did not significantly improve emotion classification. The ineffectiveness of POS features suggests that emotions are expressed more implicitly and not just by direct words (e.g. emotional adjectives). This is similar to the findings of earlier research on emotion classification [28].

On the other hand, when n-grams are combined with contextual features performance improves over the combination of n-grams and POS features. However the combination does not consistently improve emotion classification over n-grams. This clearly suggests that the simple counts of entities such as negations, emoticons, sentiment words, punctuation etc which are found effective for sentiment classification [75] cannot be directly extended for emotion classification. Finally the combination of n-grams, POS and CF also did not consistently improve emotion classification over n-grams. These experiments clearly reflect the limitations of corpus level features identified in literature (refer section 2.3.3 in chapter 2). In the following sections we discuss the results for lexicon based features and the hybrid features obtained by combining baseline and lexicon based features.

7.2.6.2 Performance of lexicon based features

Emotion classification results using lexicon based features for SemEval-07, Twitter, blogs and ISEAR data sets are shown in Figures 7.25, 7.26, 7.27 and 7.28 respectively. The x-axis in each of these figures indicate the different lexicon based features extracted using the knowledge of GPELs and DSELs (refer section 2.3.2.1 in Chapter 2). The y-axis indicates the overall performance for each feature. Overall performance is measured by combining (average) the macro-averaged F-score of all the emotion classes. Observe that TEC, TEI and MEI features consider all the words, whereas GEI and GEC features are selective. For example $GEC_{\delta 1}$ accounts only for words which have an association score with an emotion in the interval [0.25, 1]. Similarly $GEC_{\delta 2}$ and $GEC_{\delta 3}$ accounts only for words with scores in the intervals [0.5, 1] and [0.75, 1]. Further, since GPELs are simple word-emotion lists (refer Table 3.1), they are limited to extract only the TEC feature. However in the case of DSELs performance comparison can be made across different lexicon based features extracted using the emotion quantification knowledge offered by DSELs (refer Table 3.5).

In general features extracted from GPELs are significantly outperformed by those extracted using DSELs. The average performance improvements of all the features extracted using DSELs over those using GPELs is nearly 22%, 3% and 13% on twitter, blogs and ISEAR data sets respectively. Further the performance improvements of the proposed DSEL based features over those of the GPELs is nearly 8%, 40%, 12% and 19% on SemEval, Twitter, blogs and ISEAR data sets respectively. Essentially this confirms that GPELs are less able to capture the context

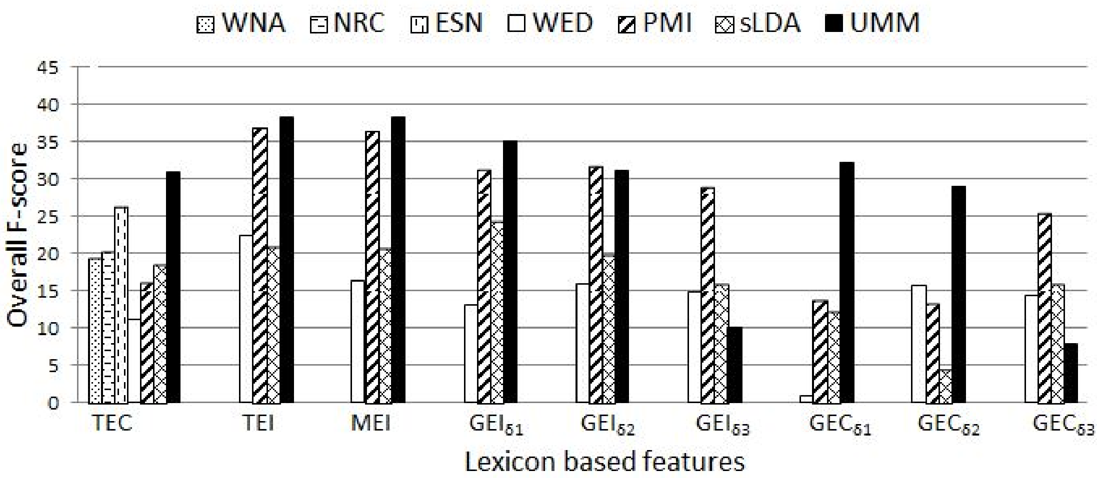


FIGURE 7.25: Overall performance on SemEval-07 with lexicon based features

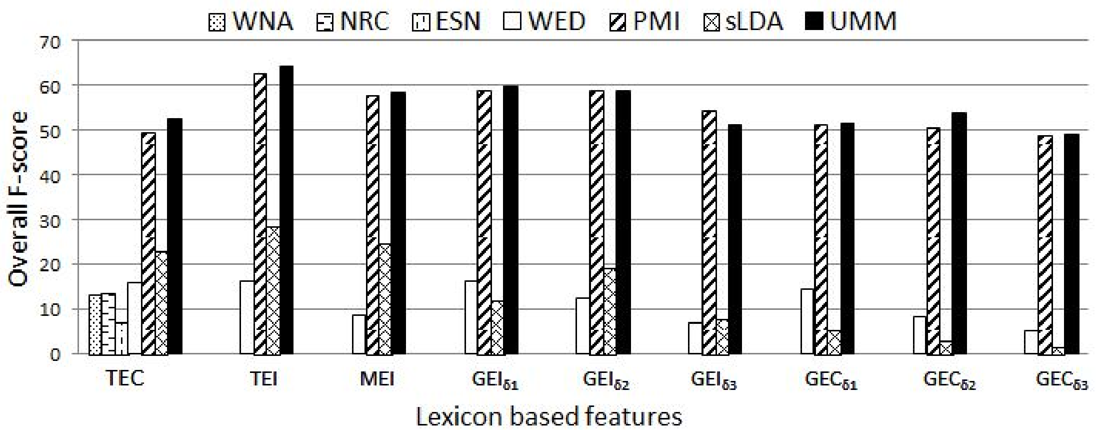


FIGURE 7.26: Overall performance on Twitter with lexicon based features

in which emotions are expressed in a domain and also are less effective to model emotions in informal text streams that typically have evolving vocabularies with time.

Comparing the results in Figures 7.25, 7.26, 7.27 and 7.28 suggest that TEI and MEI features consistently outperform GEI and GEC features. This is expected since the GEI and GEC features utilize only high intensity emotion words from a DSEL, resulting in a drop in coverage. Further a general trend of performance degradation is observed on all the data sets with GEI, GEC features as threshold values increase from δ_1 (0.25) to δ_2 (0.5) to δ_3 (0.75). This is expected since the proportion of high intensity emotion words, follow a decreasing series for increasing values of threshold from 0.25 to 0.75, resulting in a further drop in coverage. However it is extremely promising to note that the GEI and GEC features extracted from the proposed lexicon significantly outperform the TEC features extracted using the GPELs. Further the proposed DSEL based features significantly outperform those extracted using WED, PMI and sLDA. In general we noticed that the generative models assumed by sLDA and WED do not effectively model the characteristics of real-world emotional data, thereby impacting the quality of the features extracted from them. Though PMI performed the best amongst the baselines, the ability of the proposed DSEL to effectively capture the associations between words and multiple emotions resulted in quality feature extraction for documents. Whilst the other DSELs also capture the word-emotion associations, the additional ability of our DSEL to discriminate between emotional and neutral words (refer Table 4.2 in Chapter 4) improved the quality of the features extracted using its knowledge.

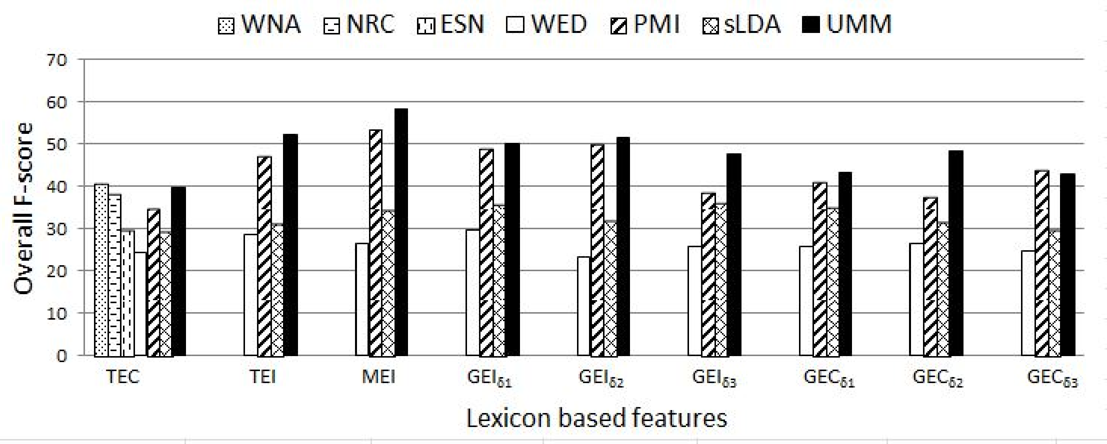


FIGURE 7.27: Overall performance on blogs with lexicon based features

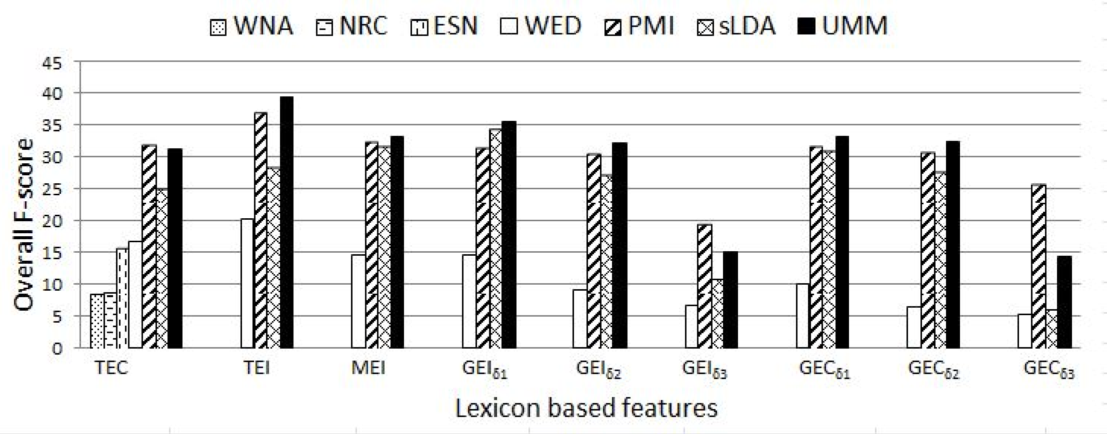


FIGURE 7.28: Overall performance on ISEAR with lexicon based features

7.2.6.3 Emotion-level performance analysis

Although the proposed DSEL in general outperformed other lexicons, we observed that the PMI lexicon is a strong competitor. Further we are also interested in comparing the performance of the lexicon based features with the baseline features discussed earlier. Accordingly we take a closer look at the baseline features[3], PMI and UMM based lexicon features by observing their performance on individual emotion classes. In particular given that not all emotions are equally complex to model, it will be useful to draw insights from those classes considered to be more challenging than others. The average F-score obtained for a class across the baseline features, lexicon based features is used as a metric to indicate its complexity. Essentially the lower the F-score, the more complex (challenging) is the class prediction.

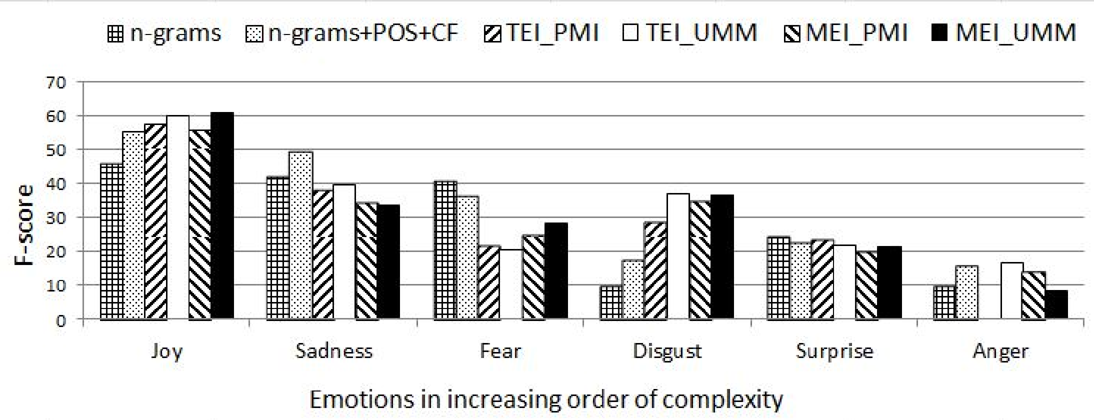


FIGURE 7.29: Emotion-level performance of different features on SemEval-07

Figures 7.29 to 7.32 capture the emotion-level performance of baseline and lexicon based features. Here the x-axis plots the results in the order of increasing emotion complexity for each data set. The y-axis indicates the performance (macro-averaged F-score). In general the results suggest that the proposed UMM lexicon outperforms the PMI lexicon in classifying harder emotions. Similarly the proposed lexicon based features are observed to be superior to the baseline features in discriminating harder emotions on twitter and ISEAR data sets. However the performance of the proposed lexicon based features were challenged on blogs, which is explained by the skewed class distribution (see Table 3.6) and on SemEval-07, where there is very limited data for learning lexicons (see Table 3.6). Nevertheless the ability to have better or comparable performance to the baseline features with significantly fewer dimensions ($|E|$, where $|E|$ is the

[3]We consider the best performing baseline features for this study

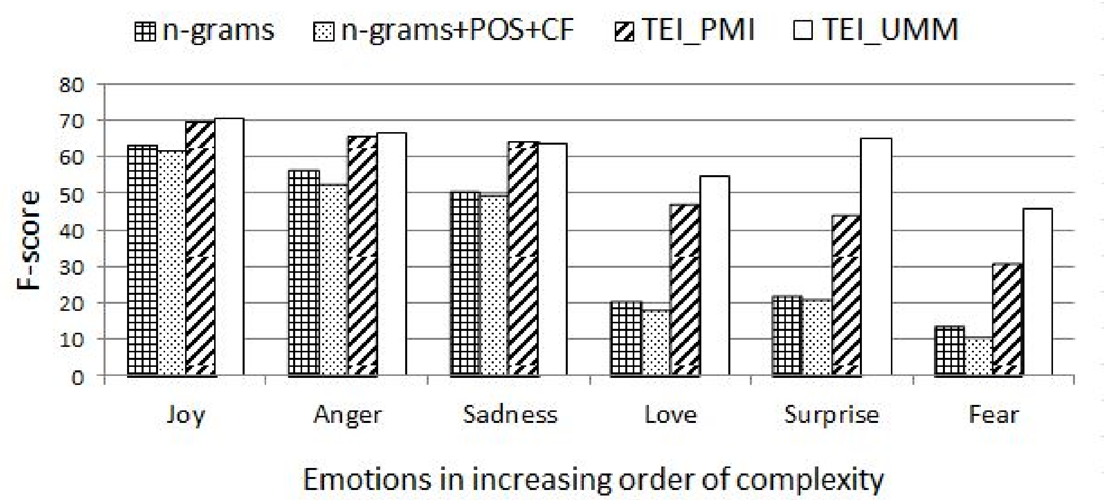


FIGURE 7.30: Emotion-level performance of different features on Twitter

number of emotion classes in a data set) is clearly an advantage of the lexicon based feature extraction methods proposed in this paper. In the following section we discuss the results for the hybrid features obtained by combining the baseline and lexicon based features.

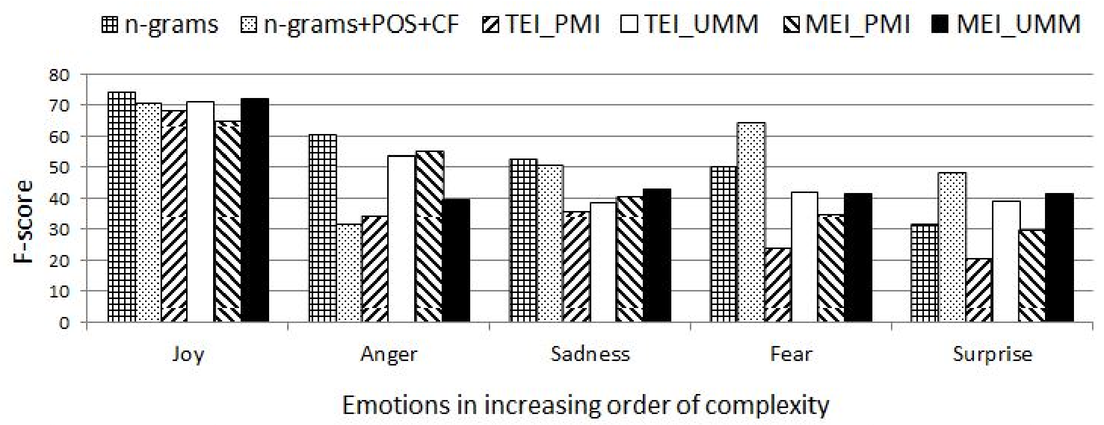


FIGURE 7.31: Emotion-level performance of different features on Blogs

7.2.6.4 Performance of hybrid features

A hybrid feature vector hyb is a $K + E$ dimensional feature vector obtained by combining a K dimensional baseline feature vector and a E dimensional lexicon based feature vector. We experimented with feature combinations of baseline[3] and lexicon based[4] features to observe for

[4]We consider the best performing lexicon based features derived using PMI, UMM for this study

Features	Test set F-Score						
	Anger	Disgust	Fear	Joy	Sadness	Surprise	Overall
SemEval-07 systems							
SWAT [89]	7.06	0.0	18.27	14.91	17.44	11.78	11.57
UA [89]	16.03	0.0	20.06	4.21	1.76	15.00	9.51
UPAR7 [89]	3.02	0.0	4.72	11.87	17.44	15.00	8.67
Baseline features							
(1) ngrams	9.37	9.54	**40.80**	45.79	41.92	24.23	35.77
(2) ngrams+POS+CF	15.42	17.40	36.52	55.32	**49.31**	22.53	**40.99**
Lexicon based features							
(3) TEI_{PMI}	0.00	28.60	21.53	57.56	38.34	24.29	36.78
(4) TEI_{UMM}	**16.78**	**36.80**	20.63	59.80	39.69	21.90	38.16
(5) MEI_{PMI}	13.86	34.85	24.67	56.00	34.32	20.00	36.54
(6) MEI_{UMM}	8.30	36.45	28.13	61.00	33.63	21.56	38.23
Hybrid features							
(1)+(3)	8.31	19.00	28.61	59.64	37.71	20.00	37.53
(1)+(4)	5.67	18.82	33.31	**60.00**	31.12	**36.40**	38.62
(1)+(5)	7.45	18.21	28.61	58.71	38.80	25.70	38.20
(1)+(6)	15.42	17.41	36.90	58.61	40.41	23.21	39.87
(2)+(3)	5.60	18.21	23.40	52.90	30.10	28.60	33.60
(2)+(4)	8.00	20.00	32.51	51.83	29.23	23.00	33.81
(2)+(5)	12.50	18.80	27.62	49.72	35.80	24.00	34.62
(2)+(6)	12.10	18.20	32.31	42.30	35.00	29.30	33.21

TABLE 7.9: Emotion classification on SemEval with hybrid features. Comparative analysis is done between systems that participated in the SemEval-07 competition, best performing baseline features, best performing lexicon based features extracted using PMI, UMM and the hybrid features

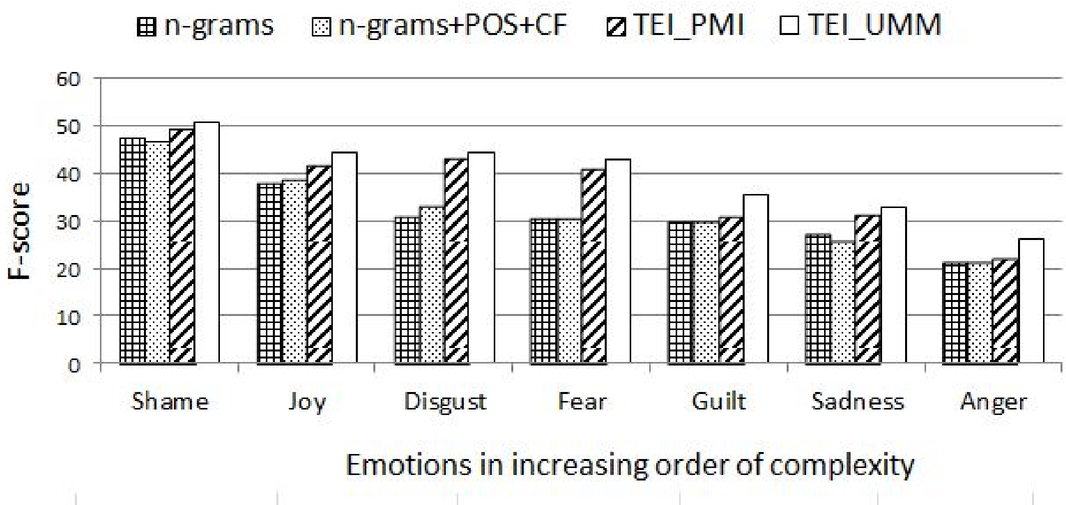


FIGURE 7.32: Emotion-level performance of different features on ISEAR

Features	Average F-Score (10-fold cross validation)						
	Anger	Fear	Joy	Sadness	Surprise	Love	Overall
Baseline features							
(1) ngrams	56.68	13.56	63.34	50.57	21.65	20.52	49.55
Lexicon based features							
(2) TEI_{PMI}	66.00	30.56	69.86	**64.42**	44.20	46.92	62.53
(3) TEI_{UMM}	**66.72**	**45.57**	**70.36**	63.67	**64.91**	**54.89**	**64.24**
Hybrid features							
(1)+(2)	56.79	31.27	61.36	45.43	28.41	24.76	49.32
(1)+(3)	59.71	27.24	67.91	54.80	33.12	31.94	55.16

TABLE 7.10: Emotion classification on Twitter with hybrid features. Comparative analysis is done between best performing baseline features, best performing lexicon based features extracted using PMI, UMM and the hybrid features

performance improvements. Emotion classification results using the hybrid features are summarized in Tables 7.9 to 7.12. Observe that performance is measured using macro-averaged F-score. We noticed that the hybrid features involving a combination of n-grams, POS, contextual features and lexicon based features deteriorates performance. We believe this is due to the ineffective contributions of POS and contextual features as discussed earlier (refer section 7.2.6.1). However the hybrid features obtained by combining n-grams and lexicon based features result in performance improvements (overall F-score) over n-grams in general, except for

Features	Average F-Score (5-fold cross validation)					
	Anger	Fear	Joy	Sadness	Surprise	Overall
Baseline features						
(1) ngrams	**60.30**	**50.04**	**73.92**	52.37	31.32	58.32
Lexicon based features						
(2) TEI_{PMI}	33.94	23.72	67.92	35.42	20.14	47.19
(3) TEI_{UMM}	53.80	41.70	71.23	38.50	38.86	52.18
(4) MEI_{PMI}	54.90	34.42	64.50	40.00	29.32	53.34
(5) MEI_{UMM}	39.32	41.63	72.29	42.68	41.54	58.16
Hybrid features						
(1)+(2)	49.00	32.00	72.10	43.80	25.00	55.20
(1)+(3)	41.56	41.70	71.62	44.90	32.16	56.46
(1)+(4)	58.60	50.00	68.90	42.40	34.10	57.78
(1)+(5)	53.72	45.53	72.78	**53.41**	**34.79**	**59.66**

TABLE 7.11: Emotion classification on Blogs with hybrid features. Comparative analysis is done between best performing baseline features, best performing lexicon based features extracted using PMI, UMM and the hybrid features

the ISEAR data set. Further the proposed UMM lexicon derived features when combined with n-grams record significant improvements over n-grams and rest of the hybrid features. Furthermore we also noticed that the hybrid features derived using the knowledge of the proposed lexicon significantly improves performance over n-grams on complex emotions such as *surprise* on SemEval; *love, surprise* and *fear* on Twitter ; and *surprise* on blogs.

7.2.7 Sentiment Ranking Results

Table 7.13 summarizes the sentiment ranking results obtained for different lexicons. In general resource-based lexicons SentiWordNet and SenticNet are outperformed by all the corpus-based lexicons. This is expected, because the vocabulary coverage of these lexicons relevant to social media is limited compared to other lexicons. Furthermore, the results also suggest that the sentiment intensity knowledge captured by the corpus-based lexicons is superior to that of resource-based lexicons.

Features	Average F-Score (5 fold cross validation)							
	Anger	Disgust	Fear	Guilt	Joy	Sadness	Shame	Overall
Baseline features								
(1) ngrams	21.11	31.00	30.62	29.85	37.86	27.24	47.56	32.19
(2) ngrams+POS+CF	21.12	33.12	30.40	29.82	38.71	25.60	46.74	32.21
Lexicon based features								
(3) TEI_{PMI}	21.86	42.92	40.76	30.93	41.56	31.30	49.48	36.96
(4) TEI_{UMM}	**25.96**	**44.52**	**42.88**	**35.42**	**44.45**	**32.66**	**50.54**	**39.48**
Hybrid features								
(1)+(3)	15.80	21.20	25.70	20.20	32.40	27.60	43.30	26.60
(1)+(4)	22.30	28.20	25.51	27.00	33.50	32.60	43.80	30.40
(2)+(3)	15.80	21.90	25.71	21.40	32.80	27.50	44.31	27.00
(2)+(4)	22.12	28.30	25.61	27.92	34.51	32.41	44.00	30.71

TABLE 7.12: Emotion classification on ISEAR with hybrid features. Comparative analysis is done between best performing baseline features, best performing lexicon based features extracted using PMI, UMM and the hybrid features

NRCHashtag lexicon performed significantly better than the remaining baselines and the proposed $EmoSentilex$. The significant performance differences between NRCHashtag lexicon and S140 lexicon and NRCHashtag lexicon and S140-UMM lexicon clearly suggests the superiority of the NRCHashtag corpus over the S140 corpus in learning transferable lexicons for sentiment intensity prediction. It would be interesting to compare the performance of these lexicons in the sentiment classification tasks.

It is extremely promising to see that the proposed lexicons outperform most of the baselines significantly. Amongst the proposed lexicons, $Sentilex$ performed significantly better than $EmoSentilex$. This is not surprising, since $Sentilex$ has the ability to incorporate the sentiment-class knowledge of the documents in the learning stage. This exactly follows the findings of earlier research in supervised and unsupervised sentiment analysis.

Method	Spearman's Rank Correlation Coefficient	p-value (at p < 0.05)
Baselines (standard sentiment lexicons)		
SentiWordNet	0.479	p < 2.64E-14
SenticNet	0.425	p < 2.64E-14
S140 lexicon	0.506	p < 2.64E-14
NRCHashtag lexicon	0.624	p < 2.64E-14
S140-UMM-lexicon	0.517	p < 2.64E-14
Proposed methods (emotion-corpus based sentiment lexicons)		
EmoSentiLex	0.572	p < 2.64E-14
Sentilex	**0.682**	n/a

TABLE 7.13: Sentiment Ranking Results

7.2.8 Sentiment Classification Results

Sentiment classification results for the S140 data set are shown in Table 7.14. Here unlike in the sentiment intensity prediction task, SentiWordNet demonstrated comparable performance with that of corpus-based lexicons. However, SenticNet does perform the worst amongst all the lexicons. This suggests that SentiWordNet is better transferable onto social media compared to SenticNet.

The S140 corpus based lexicons significantly outperform NRCHashtag lexicon, given their advantage to train on a corpus, that is similar to the test set. However, the proposed lexicon $Sentilex$ recorded the best performance on this data set. once again the superiority of $Sentilex$ over $EmoSentilex$ is evidenced, given its ability to incorporate sentiment-class knowledge of the documents in the learning stage. The performance improvements of emotion corpus based sentiment lexicons over a majority of baseline lexicons, clearly suggests that emotion knowledge when exploited effectively is very useful for sentiment analysis.

Table 7.15 summarizes the results for different lexicon on the SemEval-2013 data set. Unlike the previous, this data set has a very skewed class distribution. The impact of this is clearly reflected in the results. Majority of the lexicons recorded strong performances in classifying

Method	Positive F-score	Negative F-score	Overall F-score	p value (p < 0.05)
Baselines (standard sentiment lexicons)				
SentiWordNet	69.42	67.60	68.51	p < 8.78E-4
SenticNet	59.88	59.84	59.86	p < 8.78E-4
S140-lexicon	71.55	69.42	70.48	p < 8.78E-4
NRCHashtag-lexicon	66.66	64.75	65.70	p < 8.78E-4
S140-UMM-lexicon	**75.14**	69.36	72.25	p < 0.32
Proposed methods (emotion-corpus based sentiment lexicons)				
EmoSentiLex	67.51	71.14	69.32	p < 8.78E-4
Sentilex	72.93	**74.11**	**73.52**	n/a

TABLE 7.14: Sentiment Classification Results on S140 test data set

positive class documents. Once again SentiWordNet demonstrated that it is better transferable onto social media compared to SenticNet.

Similar to the previous data set, S140 corpus based lexicons performed better than NRCHashtag corpus based lexicon. Overall comparison across the evaluation tasks suggests that S140 corpus based lexicons record better performance in sentiment classification, whereas NRCHashtag lexicon records better performance in sentiment quantification. This offers interesting directions for future work on composing different corpora for learning sentiment lexicons.

The proposed lexicon $EmoSentilex$ performed significantly below most of the lexicons on this data set. We believe the inability to learn the document-sentiment relationships, coupled with the skewed class distribution characteristics of the data set resulted in such performance degradation. However, our proposed lexicon $Sentilex$ significantly outperformed all the remaining lexicons. The consistent performance of $Sentilex$ in all the evaluation tasks, strongly evidences the correlation between emotions and sentiments. We believe that the emotion-sentiment mapping in psychology effectively clusters the emotion corpus into sentiment classes, thereafter the ability of the UMM model to effectively capture the word-sentiment relationships resulted in the performance improvements for $Sentilex$.

Method	Positive F-score	Negative F-score	Overall F-score	p-value ($p < 0.05$)
Baselines (standard sentiment lexicons)				
SentiWordNet	80.14	50.38	65.26	p < 6.23E-3
SenticNet	54.95	55.94	55.45	p < 6.23E-3
S140-lexicon	80.13	57.87	69.00	p < 6.23E-3
NRCHashtag-lexicon	80.25	53.98	67.11	p < 6.23E-3
S140-UMM-lexicon	78.87	55.85	67.36	p < 6.23E-3
Proposed methods (emotion-corpus based sentiment lexicons)				
EmoSentiLex	64.51	48.37	56.44	p < 6.23E-3
Sentilex	**83.06**	**60.98**	**72.02**	n/a

TABLE 7.15: Sentiment Classification Results on SemEval-2013 data set

7.3 Chapter Summary

In this chapter we presented the experimental evaluation corresponding to the methods proposed in chapters 4, 5 and 6. First, we formally outlined the different evaluation tasks that assess the performance of the proposed methods in comparison with the state-of-the-art baselines. Second we presented the maximum likelihood process to empirically estimate the parameter λ of the proposed UMM method, for each emotion and also the EM iterations. We observed that for most of the emotions, the optimum value for λ is less than 1, thus indicating the noisy nature of real-world emotion data and the need for a mixture model which models both the emotionality and neutrality of documents at the word-level. In general we found emotions such as *anger, joy and sadness* have less noise compared to emotions such as *disgust, fear and surprise*. Further we also observed that the EM iterations took longer to converge on noisy-labelled documents (e.g. tweets) compared to hand labelled documents (e.g. blogs), indicating that label quality influences the learning of the EM algorithm.

Thereafter, we presented an evaluation which measures the quality of the emotion language models (emotion topics) learnt by generative methods such as sLDA and the proposed UMM method. It was observed that UMM learns topics that have significantly lower perplexity, suggesting that UMM model is better generalizable compared to that of sLDA. In order to assess the word-emotion relationships learnt by each of the DSELs, we analysed the most expressive words for an emotion identified by each DSEL. It was observed that UMM in general learns better vocabulary for each emotion compared to other DSELs. In the word-emotion classification evaluation, we observed that UMM lexicon, significantly outperforms other DSELs. We believe that the ability of the UMM method to discriminate between emotional and emotion-neutral words boosted its performance. This evaluation, further confirmed that UMM is able to learn quality word-emotion relationships compared to other DSELs. In the document-emotion ranking evaluation we observed that the proposed UMM method exhibited significant improvements over the baseline DSELs. Further on the Twitter events data set, the performance improvements observed for the proposed UMM Twitter emotion lexicon speaks for its transferable ability between domains of same genre.

A comparative analysis of emotion classification results on four benchmark data sets (news headlines, tweets, blogs and incident reports) suggests that the proposed features (refer chapter 5) extracted using the knowledge of DSELs significantly outperform those extracted from GPELs. Further the proposed features also perform significantly better over n-gram features and their combination with features based on part-of-speech information and sentiment knowledge. Closer examination of DSEL results show that the proposed features extracted from our UMM lexicon perform significantly better over those extracted using state-of-the-art methods such as PMI and sLDA on all the data sets. A deeper analysis of the results suggest that the proposed UMM lexicon features are better able to classify harder emotions such as *love, fear, anger, surprise* etc. Here the use of lexicons as a means to extract new features of very low dimensions for classification purposes is shown to be a promising strategy. These findings are very useful given the need for efficient and effective representations. Finally the hybrid features derived using the combination of n-grams and the proposed lexicon based features also resulted in consistent and significant improvements over n-gram features. This clearly confirms that a high quality lexicon which can closely capture the emotional context of a domain, when utilized effectively offers impactful knowledge for a machine learning classifier in emotion text classification.

Finally in the evaluation of emotion knowledge for sentiment analysis on Twitter, we observed that the proposed generative mixture model (UMM) when combined with the emotion-sentiment mapping proposed in psychology yield significant improvements over standard sentiment lexicons which are agnostic to the rich emotion knowledge present in an emotion-corpus. We observed consistent and significant improvements for the proposed methods which learn emotion-aware sentiment lexicons in a sentiment intensity prediction and sentiment classification tasks on benchmark data sets.

Chapter 8

Conclusions and Future Work

In this thesis, we addressed the problem of emotion detection from text using a generative mixture model based emotion lexicon that jointly models the emotionality and neutrality of words. We modelled the problem of emotion detection with a focus on variety of tasks such as word-emotion classification, word-emotion ranking and document-emotion classification. Accordingly we utilized the knowledge of the emotion lexicon to model emotion at word-level, phrase-level and document level. Further we also proposed novel lexicon-based methods that adopt an emotion-rich corpus for sentiment analysis in conjunction with the theoretical emotion-sentiment mapping proposed in psychology. The work in this thesis was aimed to achieve five research objectives. In this chapter we revisit them before drawing conclusions and pointing to future extensions of our work.

8.1 Objectives Revisited

1. *To develop an effective methodology to automatically generate a domain specific emotion lexicon (DSEL) that captures word level associations with emotions*
   General purpose emotion lexicons (GPELs), due to the static and formal nature of their vocabulary are inadequate in capturing the informal and creative expressions used in different domains to convey emotions. Especially in domains such as Twitter and internet blogs the vocabulary is constantly evolving and it is necessary to develop models that can

account for such variations in natural language expressions that convey emotions. Towards this objective we developed an expectation maximization (EM) based generative model (see Chapter 4) that can automatically extract a word-emotion lexicon from a corpus of emotion labelled documents. The uniqueness of the model lies in its ability to not just quantify the emotionality of words but also their neutrality. We compared the performance of the proposed lexicon against existing GPELs and other state-of-the-art DSELs proposed using PMI and LDA. We observed significant improvements for the proposed lexicon over the baselines in different emotion detection tasks. Further the emotion-topics generated using the proposed method had significantly lower perplexity compared to those from LDA.

2. *To utilize the knowledge of the DSEL effectively to extract lexicon based representations for emotion text classification using machine learning*

   Since a high quality DSEL captures the expressions that are emotion-rich, our objective is to leverage the availability of such DSEL to extract lexicon based features that can effectively represent text for emotion text classification. Towards this objective in Chapter 5 we have introduced novel ways in which the knowledge of a DSEL can be adopted for emotion feature extraction. We have similarly used other state-of-the-art DSELs based on PMI and LDA to extract emotion features. We observed that the proposed DSEL based emotion features performed significantly better over other DSEL based features in emotion classification tasks on benchmark data sets. This clearly illustrates the ability of the proposed DSEL to discriminate between emotion-relevant and emotion-irrelevant words thereby influencing the quality of performance of the lexicon based features extracted using it.

3. *To investigate the role of hybrid text representations obtained by combining lexicon based features and non-lexicon based features such as n-grams, POS features, sentiment features for emotion text classification using machine learning*

   Though DSELs are powerful tools for emotion detection, in the case of some domains (e.g. Twitter), all emotions are not expressed in same volumes to capture the word-emotion associations in the form of a DSEL. Therefore in Chapter 5, we investigated the role of additional knowledge such as n-gram features, POS features and sentiment features to augment DSEL based emotion features for emotion text classification. We observed that the hybrid features obtained by combining all (lexical, non-lexical) features in general did not improve performance over lexicon based features. However the hybrid features obtained by combining n-grams and lexicon based features resulted in performance improvements (overall F-score) over n-grams in general, except for the ISEAR data set. Further the proposed UMM lexicon derived features when combined with n-grams record significant improvements over n-grams and rest of the hybrid features. Furthermore we also noticed that the hybrid features derived using the knowledge of the proposed lexicon significantly improves performance over n-grams on low volume emotions such as *surprise* on SemEval; *love, surprise* and *fear* on Twitter ; and *surprise* on blogs.

4. *To study the role of emotion knowledge for sentiment analysis on social media*

   Though research in psychology defines sentiment and emotion differently [34], it also provides a relationship between them [31]. Further research in emotion classification [28, 63] demonstrated the usefulness of sentiment features extracted using a lexicon for document representation. Similarly emoticons used as features to represent documents improved sentiment classification [16, 19]. However, the exploration of emotion knowledge for sentiment analysis is limited to emoticons [19, 35, 36]. In this objective we investigate the role of an emotion corpus which captures a wide range of expressions such as emoticons, emotion hashtags, elongated words and their concatenated variants which form a relevant source of knowledge for sentiment analysis. In Chapter 6 we proposed two different methods to adopt an emotion corpus of tweets in conjunction with theoretical relationship constructs between emotions and sentiments proposed in psychology to learn emotion-aware sentiment lexicons. We compared the performance of such lexicons with standard sentiment lexicons that are emotion-agnostic. We observed that the proposed emotion-aware models significantly outperformed the baselines in different sentiment analysis tasks on benchmark Twitter data sets.

5. *To comprehensively evaluate the different methods/strategies proposed for emotion detection from text and also the methods to apply emotion knowledge for sentiment analysis*

   We conducted evaluations to ascertain the effectiveness of each of the different methods proposed in this research: EM generative model based DSEL, DSEL based emotion features, hybrid features and emotion-aware sentiment lexicons (Chapter 7). We compare the performance of the proposed methods against state-of-the-art baselines using a multitude of emotion detection tasks: *word-emotion classification, document-emotion ranking, document-emotion classification, word-sentiment ranking and document-sentiment classification*. All the performance evaluations are done using benchmark data sets gathered from different domains (social media and non-social media). We use standard evaluation metrics that are relevant for each task to compare the performance differences between proposed methods and the baselines. Finally we use t-test to quantify statistical significance of performance improvements.

8.2 Future Work

In this section we highlight some of the limitations of the work we presented in this thesis and also indicate some desirable future extensions. Firstly, since this research focusses on emotion detection from text using a lexicon based approach, a natural extension to this work is to learn multi-word-emotion lexicons (i.e. bigram and trigram) following the recent trend in multi-word sentiment and emotion detection [100]. Also the knowledge of the proposed DSEL can be used in conjunction with knowledge bases such as SenticNet and EmoSenticNet to extract effective features to represent documents for emotion classification. Secondly, the work presented in this thesis, aimed at capturing and quantifying word-emotion associations in the form of a lexicon can be adopted for dynamic and evolving streams of data on social media (e.g. Twitter). In particular efficient methods to adjust the emotion scores of new words that are encountered in the dynamic streams without having to re-learn or re-train the lexicon would be a very useful research direction given the characteristics of social media big data such as velocity and veracity. Thirdly, the emotion features extracted using the knowledge of the lexicon can be enhanced by augmenting with knowledge from dense representation of text such as word embeddings [101] and [102]. Given the recent success of neural network based dense text representations for different natural language processing tasks such as *sentiment analysis* [103], text classification [104] etc it is interesting to investigate for methods that can inject the knowledge captured by an emotion lexicon into the word embeddings to make them more emotion specific. Finally, as we investigated the role of emotion knowledge for sentiment analysis, it is also imperative to design models that can jointly model both sentiment and emotion simultaneously. This will further help in understanding the manner in which sentiment and emotion occur in real world data, thereby making it possible to validate the theoretical relationships proposed between them in psychology more comprehensively.

Bibliography

[1] Binali H, Potdar V. Emotion detection state of the art. In: Proceedings of the CUBE International Information Technology Conference. CUBE '12. New York, NY, USA: ACM; 2012. p. 501–507. Available from: http://doi.acm.org/10.1145/2381716.2381812.

[2] Shaikh PHIM M A M. 4. In: Processing AI, editor. A Linguistic Interpretation of the OCC Emotion Model for Affect Sensing from Text; 2009. p. 45–73.

[3] Pang B, LL. Opinion mining and sentiment analysis. Foundations and Trends in Information Retrieval. 2008;2(1):1–135.

[4] Strapparava C, Valitutti A. WordNet-Affect: An affective extension of Wordnet. In: Proc of LREC, pp. 1083-1086; 2004. .

[5] Mohammad SM, Turney P. Crowdsourcing a Word-Emotion Association Lexicon. Computational Intelligence, 29(3), pp 436-465. 2013;.

[6] Poria S, Gelbukh A, Hussain A, Das D, Bandopadhyay S. Enhanced SenticNet with Affective Labels for Concept-based Opinion Mining. IEEE Intelligent Systems, Vol 28, issue 2, pp 31-38. 2013;.

[7] Pang B, Lee L. Opinion Mining and Sentiment Analysis. Foundations and Trends in Information Retrieval. 2008;2(1):1–135.

[8] Liu B. Handbook of Natural Language Processing. In: Indurkhya N, Damerau FJ, editors. Sentiment Analysis and Subjectivity. 2nd ed. Chapman and Francis; 2010. p. 627–666.

[9] Hu M, Liu B. Mining and Summarizing Customer Reviews. In: Proc of the ACM SIGKDD International Conference on Knowledge Discovery & Data Mining; 2004. .

[10] Stone P, Dunphy D, C S, M S O, associates. The General Inquirer: A Computer Approach to Content Analysis. The MIT Press. 1966;.

[11] Wilson T, Wiebe J, Hoffmann P. Recognizing Contextual Polarity in Phrase-Level Sentiment Analysis. In: Proc. of HLT-EMNLP-2005; 2005. .

[12] Esuli A, Baccianella S, Sebastiani F. Sentiwordnet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining. In: Proc of LREC; 2010. .

[13] Cambria E, Olsher D, Rajagopal D. SenticNet 3: A Common and Common-Sense Knowledge Base for Cognition-Driven Sentiment Analysis. In: 28th AAAI conf on Artificial Intelligence, pp. 1515-1521; 2014. .

[14] Fellbaum, Christiane. WordNet and wordnets. In: Brown, Keith et al (eds), Encyclopedia of Language and Linguistics. 2005;Second Edition, Oxford: Elsevier,:665–670.

[15] Liu H, Singh P. ConceptNet- A practical Commonsense Reasoning Tool-kit. BT Technology Journal, 22(4), pp 211-226. 2004;.

[16] Mohammad SM, Kiritchenko S, Zhu X. NRC-Canada: Building the State-of-the-Art in Sentiment Analysis of Tweets. In: 7th International Workshop on Semantic Evaluation (SemEval 2013), pp 321-327; 2013. .

[17] Feng S, Song K, Wang D, Yu G. A word-emotion mutual reinformcement ranking model for building sentiment lexicon from massive collection of microblogs. World Wide Web, 18(4):949-967. 2015;.

[18] Alec Go RB, Huang L. Twitter Sentiment Classification using Distant Supervision. Processing. 2009;.

[19] Hogenboom A, Bal D, Frasincar F, Bal M. Exploiting Emoticons in Polarity Classification of Text. Journal of Web Engineering. 2013;.

[20] Mitchell TM. Machine Learning. McGraw Hill; 1997.

[21] Calvo R, A D'Mello S. Affect Detection: An Interdisciplinary Review of Models, Methods, and Their Applications. IEEE Transactions on Affective Computing. 2010;Volume: 1 , Issue: 1:18– 37.

[22] Ekman P. An argument for basic emotions. Cognition and Emotion, 6(3), pp 169-200. 1992;.

[23] Parrott W. Emotions in Social Psychology. Psychology Press, Philadelphia. 2001;.

[24] Plutchik R. A general psychoevolutionary theory of emotion. In R Plutchik & H Kellerman (Eds), Emotion: Theory, research, and experience:. 1980;Vol. 1. Theories of emotion (pp. 3-33). New York: Academic:(pp. 3–33).

[25] Gunes H, Pantic M. Automatic, Dimensional and Continuous Emotion Recognition. International Journal of synthetic emotions. 2010;vol. 1, no. 1,:pp. 68–99.

[26] Picard RW. Affective Computing; 1997.

[27] Ortony TTJ A. What's basic about basic emotions? Psychological Review. 1997;p. 315–331.

[28] Wang W. Harnessing Twitter "Big Data" for Automatic Emotion Identification. In: Privacy, Security, Risk and Trust (PASSAT), 2012 International Conference on and 2012 International Confernece on Social Computing (SocialCom); 2012. .

[29] Qadir A, Riloff E. Bootstrapped Learning of Emotion Hashtags #hashtags4you. In: the 4th Workshop on Computational Approaches to Subjectivity, Sentiment & Social Media Analysis (WASSA 2013); 2013. .

[30] Aman S, Szpakowicz S. Identifying expressions of emotion in text. In: Proceedings of the 10th international conference on Text, speech and dialogue; 2007. .

[31] Binali V H Potdar, Wu C. Computational approaches for emotion detection in text. In: 4th IEEE International Conference on Digital Ecosystems and Technologies DEST; 2010. .

[32] Jin X, Wang Z. An emotion space model for recognition of emotions in spoken chinese. In: Proceedings of the First international conference on Affective Computing and Intelligent Interaction; 2005. .

[33] BenZe A. The Subtlety of Emotions. Cambridge, MA, USA: MIT Press. 2000;vol 12.

[34] Munezero M, Montero CS, Sutinen E, Pajunen J. Are They Different? Affect, Feeling, Emotion,Sentiment, and Opinion Detection in Text. IEEE Transactions on Affective Computing, Vol 5 No 2. 2014;.

[35] Hu X, Tang J, Gao H, Liu H. Unsupervised Sentiment Analysis with Emotional Signals. In: Proceedings of the International World Wide Web Conference (WWW); 2013. .

[36] Jiang F, Liu YQ, Luan HB, Sun JS, Zhu X, Zhang M, et al. Microblog Sentiment Analysis with Emoticon Space Model. Journal of Computer Science and Technology, vol 30(5), pp 1120-1129. 2015;.

[37] Elliott C. The Affective Reasoner: A Process Model of Emotions in a Multi-agent System, PhD thesis. Northwestern University, The Institute for the Learning Sci-ences, Technical Report No. 32; 1992.

[38] Andrew Ortony GLC, Collins A. The Cognitive Structure of Emotions. Cambridge University Press; 1988.

[39] Hu M, BL. "Mining and summarizing customer reviews.". In: Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD-2004, full paper), Seattle, Washington, USA, Aug 22-25, 2004; 2004. .

[40] Theresa Wilson JW, Hoffmann P. Recognizing Contextual Polarity in Phrase-Level Sentiment Analysis. In: Proc. of HLT-EMNLP-2005; 2005. .

[41] Cambria E, Poria S, Bajpai R, Schuller B. SenticNet 4: A semantic resource for sentiment analysis based on conceptual primitives. In: COLING, Osaka (2016); 2016. .

[42] Chen Y, Skiena S. Building Sentiment Lexicon for all major loanguages. In: Proc of the 52nd Annual Meeting of the Assoc for Computational Linguistics, pp. 383-389; 2014. .

[43] Song K, Feng S, Gao W, Wang D, Chen L, Zhang C. Build Emotion Lexicon from Microblogs by Combining Effects of Seed Words and Emoticons in a Hetereogeneous Graph. In: Proc of the 26th ACM Conference on Hypertext & Social Media, pp. 283-292; 2015. .

[44] Bao S, Xu S, Zhang L, Yan R, Su Z, Han D, et al. Mining Social Emotions from Affective Text. In: IEEE Trans. Knowledge and Data Eng. 24(9), pp. 1658-1670; 2012. .

[45] Muhammad A, Wiratunga N, Lothian R. Contextual Sentiment Analysis for Social media Genres. Knowledge-Based Systems. 2016;108:92–101.

[46] Rao Y, Lei J, Wenyin L, Li Q, Chen M. Building Emotional Dictionary for Sentiment Analysis of Online News. World Wide Web, Vol 17, pp 723-742. 2014;.

[47] Staiano J, Guerini M. DepecheMood: a Lexicon for Emotion Analysis from Crowd-Annotated News. In: Proc of the 52nd Annual Meeting of the Assoc for Computational Linguistics, pp. 427-433; 2014. .

[48] Mohammad SM. #Emotional Tweets. In: Proc of The First Joint Conference on Lexical and Computational Semantics, pp. 246-255; 2012. .

[49] Yang C, Lin KHY, Chen HH. Emotion Classification Using Web Blog Corpora. In: Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence. WI '07; 2007. .

[50] Yang M, Peng B, Chen Z, Dingju Zhu aKC. A Topic Model for Building Fine-grained Domain-specific Emotion Lexicon. In: Proc of the 52nd Annual Meeting of the Assoc for Computational Linguistics, pp. 421-426; 2014. .

[51] Mcauliffe JD, Blei DM. Supervised Topic Models. In: Advances in Neural Information Processing Systems, pp. 121-128; 2007. .

[52] Deepak P, Visweswariah K. Unsupervised solution post identification from discussion forums. In: ACL; 2014. .

[53] Deepak P, Visweswariah K, Wiratunga N, Sani S. Two-part segmentation of text documents. In: CIKM; 2012. .

[54] Strapparava C MR. Annotating and Identifying Emotions in Text. In: Intelligent Information Access; 2010. .

[55] Ameeta Agarwaal AA. Unsupervised Emotion Detection from Text Using Semantic and Syntactic Relations. In: 2012 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology; 2012. .

[56] Alm CO, Roth D, Sproat R. Emotions from text: machine learning for text-based emotion prediction. In: Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing. HLT '05. Stroudsburg, PA, USA: Association for Computational Linguistics; 2005. p. 579–586. Available from: http://dx.doi.org/10.3115/1220575.1220648.

[57] Matsumoto D, Kudoh T, Scherer KR, Wallbott HG. Antecedents of and reactions to emotions in the US and Japan. Journal of Cross-Cultural Psychology. 1988;19:267–286.

[58] Smekal RBJKZ. Recognition of Emotions in Czech Newspaper Headlines. In: Radioengineering Vol 20 pp 39-47; 2011. .

[59] Pang B, Lee L, Vaithyanathan S. Thumbs up?: Sentiment Classification using Machine Learning Techniques. In: ACL-02 Conference on Empirical Methods in Natural Language Processing, Vol 10; 2002. .

[60] Aman S, Szpakowicz S. Using Roget's Thesaurus for Fine-grained Emotion Recognition. In: International Joint Conference on Natural Language Processing; 2008. .

[61] Purver M, Battersby S. Experimenting with distant supervision for emotion classification. In: Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics; 2012. .

[62] Albornoz JCD, Plaza L, Gervas P. Improving Emotional Intensity Classification using Word Sense Disambiguation. Special issue: Natural Language Processing and its Applications Journal on Research in Computing Science. 2010 03/2010;46:131 – 142.

[63] Ghazi D, Inkpen D, Szpakowicz S. Hierarchical approach to emotion recognition and classification in texts. In: Proceedings of the 23rd Canadian conference on Advances in Artificial Intelligence; 2010. .

[64] Mohammad SM. Portable Features for Classifying Emotional Text. In: Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 587-591, Montreal , Canada; 2012. .

[65] Jarmasz M, Szpakowicz S. The Design and Implementation of an Electronic Lexical Knowledge Base. In: In Proceedings of the 14th Biennial Conference of the Canadian

Society for Computational Studies of Intelligence (AI 2001), Ottawa, Canada, June 325-333; 2001. .

[66] K Roberts JJJG M A Roach, Harabagiu SM. "EmpaTweet: Annotating and Detecting Emotions on Twitter",. In: in Proc. LREC, 2012, pp.3806-3813.; 2012. .

[67] Qadir A, Riloff E. Bootstrapped Learning of Emotion Hashtahs #hashtags4you. In: In the 4th Workshop on Computational Approaches to Subjectivity, Sentiment & Social Media Analysis (WASSA 2013); 2013. .

[68] Ahmed A A Esmin dOJ Roberto L, Matwin S. Hierarchical Classification Approach to Emotion Recognition in Twitter. In: 2012 11th International Conference on Machine Learning and Applications; 2012. .

[69] Boyd D, Golder S, Lotan G. Tweet, Tweet, Retweet: Conversational Aspects of Retweeting on Twitter. In: Proceedings of the 2010 43rd Hawaii International Conference on System Sciences, Washington, DC, USA; 2010. .

[70] Poria S, Gelbukh A, Cambria E, Hussain A, Huang GB. EmoSenticSpace: A Novel Framework for Affective Common-Sense Reasoning. Knowledge-Based Systems 69, pp 108-123. 2014;.

[71] Andrew Ortony GLC, Collins A. The Cognitive Structure of Emotions. Cambridge University Press; 1988.

[72] Blei DM, Ng AY, Jordan MI. Latent dirichlet allocation. the Journal of machine Learning research. 2003;3:993–1022.

[73] Mei Q, Ling X, Wondra M, Su H, Zhai C. Topic sentiment mixture: modeling facets and opinions in weblogs. In: Proceedings of the 16th international conference on World Wide Web. ACM; 2007. p. 171–180.

[74] Lin C, He Y. Joint sentiment/topic model for sentiment analysis. In: Proceedings of the 18th ACM conference on Information and knowledge management. ACM; 2009. p. 375–384.

[75] Mohammad SM, Kiritchenko S, Zhu X. NRC-Canada: Building the State-of-the-Art in Sentiment Analysis of Tweets. In: Seventh International Workshop on Semantic Evaluation (SemEval 2013), pages 321-327; 2013. .

[76] Gimpel K, Schneider N, O'Connor B, Das D, Mills D, Eisenstein J, et al. Part of Speech Tagging for Twitter: Annotation, Features, and Experiments. In: Annual Meeting Of the Association for Computational Linguistics; 2011. .

[77] Suttles J, Ide N. Distant Supervision for Emotion Classification with Discrete Binary Values. Computational Linguistics and Intelligent Text Processing, 2013. 2013;.

[78] Thelwall M, Buckley K, Paltoglou G. Sentiment Strength Detection for the Social Web. Journal of the American Society for Information Science and Technology, 63(!), 163-173. 2012;.

[79] Burges CJC. A Tutorial on Support Vector Machines for Pattern Recognition. Data Mining and Knowledge Discovery. 1998;2:121–167.

[80] Hsu CW, Lin CJ. A Comparison of Methods for Multiclass Support Vector Machines. Transactions Neural Networks. 2002;.

[81] R-E Fan CJHXRW K-W Chang, Lin CJ. LIBLINEAR: A library for Large Linear Classification. Journal of Machine Learning Research 9(2008), 1871-1874;.

[82] Joachims T. Text Categorization with Suport Vector Machines: Learning with Many Relevant Features. In: Proceedings of the 10th European Conference on Machine Learning, ECML; 1998. .

[83] Joachims T. Learning to Classify Text Using Support Vector Machines: Methods, Theory and Algorithms. Kluwer Academic Publishers; 2002.

[84] Bilmes J. A Gentle Tutorial of the EM Algorithm and its application to Parameter Estimation for Gaussian Mixture and Hidden Markov Models. Technical Report, University of Berkeley; 1997.

[85] Dempster AP, Laird NM, Rubin DB. Maximum Likelihood from Incomplete Data via the EM-algorithm plus discussions on the paper. Journal of the Royal Statistical Society, 39(B): 1-38. 1977;.

[86] Buitinick L. A note on the EM algorithm for Parismonious Language Models. Infirmatics Institute; 2013.

[87] Hiemstra D, Robertson S, Zaragoza H. Parsimonious Language Models for Information Retrieval. In: Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval; 2004. .

[88] Deepak P, Visweswariah K. Unsupervised solution post identification from discussion forums. In: ACL; 2014. .

[89] Strapparava C, Mihalcea R. SemEval-2007 Task 14: Affective Text. In: In Proceedings of the 4th International Workshop on Semantic Evaluations (SemEval-2007), pages 70-74, Prague, June; 2007. .

[90] Vu HT, Neubig G, Sakti S, Toda T, Nakamura S. Acquiring a Dictionary of Emotion-Provoking Events. In: Proceedings of the 14th European Chaper of the Association for Computational Linguistics, Gothenburg, Sweden, April; 2014. .

[91] Nakov P, Rosenthal S, Kozareva Z, Stoyanov V, Ritter A, Wilson T. SemEval-2013 Task2: Sentiment Analysis in Twitter. In: Proceedings of the 7th International Workshop on Semantic Evaluation (SemEval-2013); 2013. .

[92] Rosenthal S, Nakov P, Kiritchenko S, Mohammad SM, Ritter A, Stoyanov V. SemEval-2015: Sentiment Analysis in Twitter. In: Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval-2015); 2015. .

[93] Sokolova M, Lapalme G. A Systematic Analysis of Performance Measures for Classification Tasks. Elsevier Information Processing and Management. 2008;.

[94] Sebastiani F. Machine Learning in Automated Text Categorization. ACM Computing Surveys. 2002;34:1–47.

[95] Wallach HM. Topic modeling: beyond bag-of-words. In: Proceedings of the 23rd international conference on Machine learning. ACM; 2006. p. 977–984.

[96] Scott S, Matwin S. Feature engineering for text classification. In: ICML. vol. 99; 1999. p. 379–388.

[97] Cormack GV. Feature engineering for mobile (SMS) spam filtering. In: In 30th ACM SIGIR Conference on Research and Development on Information Retrieval; 2007. .

[98] De Vel O, Anderson A, Corney M, Mohay G. Mining e-mail content for author identification forensics. ACM Sigmod Record. 2001;30(4):55–64.

[99] Park S, Lee W, Moon IC. Efficient extraction of domain specific sentiment lexicon with active lea. Elsevier Pattern Recognition Letters. 2015;.

[100] Cambria E. Affective Computing and Sentiment Analysis. IEEE Intelligent Systems 31(2), pp 102-107. 2016;.

[101] Mikolov T, Sutskever I, Chen K, Corrado GS, Dean J. Distributed representations of words and phrases and their compositionality. In: Advances in NIPS; 2013. p. 3111–3119.

[102] Lev G, Klein B, Wolf L. In defense of word embedding for generic text representation. In: Inter. Conf. on Applications of Natural Language to Information Systems. Springer; 2015. p. 35–50.

[103] Tang D, Wei F, Yang N, Zhou M, Liu T, Qin B. Learning Sentiment-Specific Word Embedding for Twitter Sentiment Classification. In: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics pp 15551565; 2014. .

[104] Armand Joulin aPBaTM and Edouard Grave. Bag of Tricks for Efficient Text Classification. CoRR. 2016;abs/1607.01759.