

KNOWLEDGE MEDIA

KMi

I N S T I T U T E

Sentiment Analysis of Microblogs

Mining the New World

Technical Report KMI-12-2
March 2012

Hassan Saif



Abstract

In the past years, we have witnessed an increased interest in microblogs as a hot research topic in the domain of sentiment analysis and opinion mining. Through platforms like Twitter and Facebook, millions of status updates and tweet messages, which reflect people's opinions and attitudes, are created and sent every day. This has recently brought great potentials and created unlimited opportunities where companies can detect the level of satisfaction or intensity of complaints about certain products and services and policy makers and politicians are able to detect the public opinions about their policies or political issues.

Sentiment analysis of microblogs faces several major challenges due to the unique characteristics possessed by microblogging services. One challenge is data sparsity. This is because microblogs contain a large number of irregular and ill-formed words due to the length limit. Another challenge is open-domain where users can post about any topic. This forces building sentiment classifier that work independently of the studied domain. Another serious challenge is data dynamics and evolution as microblogs are produced continuously by a large and uncontrolled number of users. This poses very strict constraints where microblogging data should be processed and analysed in real-time.

This report summarises the previous work in microblog sentiment analysis and discusses the major challenges that are yet to be overcome. It then presents my pilot work that has been undertaken so far in which I proposed a novel feature-approach to address the data sparsity problem of tweets data. The future plan for the remaining two years is given at the end of the report.

Keywords: Sentiment analysis, Microblogging Services, Semantic Smoothing, Political Tweet Analysis, Twitter, Facebook.

To the olden days of Damascus..

Contents

1	Introduction	6
1.1	The Phenomenon of Microblogs	6
1.2	Motivation	7
1.3	3D Problems	7
1.4	Research Objectives	8
1.5	First Touch	9
1.6	Report Overview	11
2	Literature Survey	12
2.1	General Overview	12
2.2	Twitter Sentiment Analysis	12
2.3	Discussion	14
3	Pilot Work	16
3.1	Alleviating Data Sparsity for Twitter SA	16
3.1.1	Twitter Sentiment Corpus	17
3.1.2	Semantic Features	18
	Semantic Concept Extraction	18
	Incorporating Semantic Concepts into NB Training	18
3.1.3	Sentiment-Topic Features	19
3.1.4	Experimental Results	21
	Pre-processing	21
	Semantic Features	22
	Sentiment-Topic Features	22
	Comparison with Existing Approaches	23
	Discussion	23
3.2	Political Tweets Sentiment Analysis	25
3.2.1	Objective	25
3.2.2	Political Tweet Corpus	25
3.2.3	Tweet Sentiment Analysis	26
3.2.4	Discussion	28
3.3	Tweenator	29
3.4	Moody	31
3.4.1	How does Moody work?	31
3.4.2	Two good things about Moody	31

4	Future Work	33
4.1	Culture-Aware Sentiment Analysis	34
4.1.1	Background	34
4.1.2	Approach	34
4.2	Progress Plan	36

List of Figures

1.1	Popularity of Leading Social Networking Sites	7
2.1	The coverage of previous work regarding about the 3D problems	15
3.1	Word frequency histogram.	16
3.2	Incorporating semantic concepts for sentiment classification.	17
3.3	Classification accuracy vs. number of topics.	23
3.4	Classification accuracy vs. number of features selected by information gain. . .	24
3.5	The tweets volume distribution for various parties.	26
3.6	The Joint Sentiment-Topic (JST) model and the modified JST with side information incorporated.	27
3.7	Sentiment distributions of the three main UK parties.	28
3.8	Tweenator as a sentiment annotation tool	29
3.9	Sentiment Plug-in Interfaces in Tweenator	30
4.1	Culture-based sentiment model	35
4.2	Main Tasks	36

List of Tables

3.1	Top 5 concepts with the number of their associated entities.	18
3.2	Extracted polarity words by JST.	20
3.3	The effect of pre-processing.	21
3.4	Sentiment classification results on the 1000-tweet test set.	22
3.5	Sentiment classification results on the original Stanford Twitter Sentiment test set.	23
3.6	Social influence ranking results.	26
4.1	Progress Plan for the Second and Third Year	37

Chapter 1

Introduction

Sentiment analysis aims to identify and extract opinions and attitudes from a given piece of text towards a specific subject. There has been much progress on sentiment analysis of conventional text, which is usually found in open forums, blogs and the typical review channels. However, sentiment analysis of microblogs is considered as a much harder problem due the unique characteristics possessed by microblogs (e.g. short length of status updates and language variations). This report studies existing literature on sentiment analysis of microblogs, raises my research questions, presents the work that have been done in the first year, and finally outlines future plan for the remaining two years.

1.1 The Phenomenon of Microblogs

Microblogging is a network service, which allows users to post and broadcast messages to other subscribed users of the same service. Microblogging services differ from traditional blogging services in that their posts are brief (typically 140 - 200 characters). The first microblogging service was *tumblelogs*, which appeared in 2005. Later years have shown a birth of different microblogging websites and services such as Twitter, Tumbler, Jaiku and Pownce (2007) and Plurk (2008). Other social media tools like Facebook, MySpace, LinkedIn, and XING also provide microblogging services, which are known in this case as *status updates*. Recent statistics as shown in Figure 1.1 show that Twitter and Facebook are now considered as the most popular social networks and microblogging services. While Twitter has 200 million users, Facebook has 800 million active users¹. 600 tweet messages and 700 status updates are sent and published every second.

Twitter is an online microblogging service, which was created in March 2006. It enables users to send and read text-based posts, known as tweets, with the 140-character limit for compatibility with SMS messaging. Twitter allows users to subscribe (called *following*) to other users' tweets. A user can forward or retweet other users' tweets to his followers (e.g. “*RT @username [msg]*” or “*via @username [msg]*”).

Facebook is an online social network, launched in February 2004. Once users register with Facebook, they can create their own personal profiles, construct their friendships networks by

¹<https://www.facebook.com/press/info.php?statistics>

adding other users as *friends*, share and exchange short textual updates known as *status updates* with the 420-character limit. Moreover, users can join groups with common interests. These groups are usually organized by private or public parties.

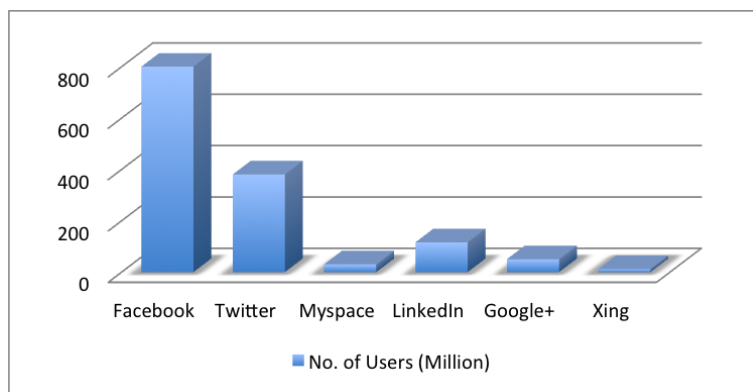


Figure 1.1: Popularity of Leading Social Networking Sites

1.2 Motivation

The emergence of social media combined with microblogging services' easy-to-use features have dramatically changed people's life with more and more people sharing their thoughts, expressing opinions, and seeking for support on such open social and highly connected environments.

Monitoring and analysing opinions from social media provides enormous opportunities for both public and private sectors. For private sectors, it has been observed [44, 41] that the reputation of a certain product or company is highly affected by rumours and negative opinions published and shared among users on social networks. Understanding this observation, companies realize that monitoring and detecting public opinions from microblogs leading to building better relationships with their customers, better understanding of their customers' needs and better response to changes in the market.

For public sectors, recent studies [4, 16] show that there is a strong correlation between activities on social networks and the outcomes of certain political issues. For example, Twitter and Facebook were used to organise demonstrations and build solidarity during Arab Spring of civil uprising in Egypt, Tunisia, and currently in Syria. One week before Egyptian president's resignation the total rate of tweets about political change in Egypt increased ten-fold. In Syria, the amount of online content produced by opposition groups in Facebook increased dramatically. Another example is the UK General Election 2010. It has been shown that activities at Twitter are a good predictor of popularities of political parties [14]. Thus tracking and analysing users' activities on social media are they key to understanding and predicting public opinions towards certain political event.

1.3 3D Problems

Much work has been done on sentiment analysis. Most of this work focuses on extracting sentiments from text found in traditional online media, such as open forums, blogs and peer-

to-peer networks. However, applying previous approaches to detect sentiment from microblogs poses new challenges due to several unique characteristics possessed by microblogs. These challenges can be categorised into three main categories as follows:

Data One common characteristic shared between many microblogging services is the short length of their update messages. While Facebook has a limit of 420 characters for status updates, Twitter has a 140-character limit. Another characteristic is language variations. Users post from different media including their personal laptops, cell phones and tablets in such, they tend to use a large variety of short forms and irregular words. These two characteristics induce significant data sparseness and thus affect the performance of typical sentiment classifiers learned from such noisy data.

Domain Microblogs like Twitter and Facebook are open social environments where there are no restrictions on what users can tweet about and in which domain. This differs from previous work on sentiment analysis, which focused on a specific domain of interest such as product reviews. Most of previous work on microblog sentiment analysis follows the supervised machine learning approaches (see Chapter 2 for a literature survey). However, supervised classifiers require training labelled data which is impractical and time-consuming to get. Also models trained on one domain might face a serious loss in performance when shifting to another domain. To overcome these drawbacks, an automated method using emoticons (called distant supervision) [11, 3, 23] was proposed. However, learning sentiment classifiers from noisy labels may hinder the overall performance. Also it is not possible to capture the sentiment of instances (e.g. tweets) with no associated emoticons. Moreover, different emotions may be associated with the same instance, which also makes distant supervision infeasible process.

Dynamics Microblogging services in general operate on the data stream fashion where data is transferred, viewed and discarded immediately. This raises new problems for sentiment classification. First, classifiers should work with limited resources of time and space. Second, the dynamic nature of the data means that we need to deal with imbalanced classes where training instances in some classes are significantly less than those in other classes. For example, a training corpus may contain more positive tweet messages than negatives ones. This also differs from the previous work which assumes the balance between negative and positive instances.

1.4 Research Objectives

It can be realized from the aforementioned problems that sentiment analysis of Microblogs faces the following challenges:

- The short length of status updates coupled with their noisy nature makes the data very sparse to analyse using standard machine learning classifiers.
- The lack of labelled data needed for classifiers training.
- Open nature of microblogs poses an open-domain problem where classifiers should work in a multi-domain environment.

- The streaming fashion of microblogs where data arrives at a high speed. This means data should be processed in real time and classifiers should adapt quickly with the newly arrived data.

Thus, the main research question of my research can be formulated as follows:

“How to build a domain-independent sentiment classifier learned from short textual sparse data, which is able to operate in a streaming fashion, and adapt dynamically to the new data.”

We can look at our research problem from a software engineering perspective, where the research question here can serve as the main functional requirement of the system. This actually helps us to frame our work and formulate our research objectives as follows:

Obj1. Data sparsity should be alleviated; this implies that data should be pre-processed before it is getting fed into classifier training.

Obj2. Sentiment classifiers should be trained in a domain-independent way. In simple words, they should be able to provide a similar performance when a domain shift occurs.

Obj3. Sentiment classifiers should be able to operate on the data streaming paradigm of microblogs. This means they should have the ability to work with limited resources of time and space.

Obj4. The problem of imbalanced sentiment distribution (sentiment drift) should be considered when building sentiment classifiers. This means classifiers are expected to work with imbalanced numbers of training instances in different classes.

Obj5. Classifiers should be easily adapted to work with different microblogging services.

In my study here, Twitter was used as a case study. The reasons behind this choice are: (1) Twitter has been used as case study by almost all previous work on sentiments analysis of microblogs. Thus conducting our experimental work on tweets data allows us to compare our approach to existing approaches. (2) Twitter is more flexible in its privacy policy than other microblogging services. For example, Twitter provides a set of APIs, which makes collecting and analysing data an easier process compared to other microblogging services such as Facebook.

It is worth mentioning that the approaches that will be proposed in my study are equally applicable to other microblogging services since the problems to be addressed are common to most other microblogging data.

1.5 First Touch

By the time of writing these lines, it has been 8 months since I started investigating sentiment analysis of microblogs. The following work has been conducted:

Literature Review I started my literature review by looking first at the problem of sentiment analysis in general, focusing later on the problem of sentiment analysis of microblogs. In order to understand the issues and major challenges faced my problem, I grouped previous work into several categories based on the proposed approaches, datasets and evaluation methodologies. This helps me to cross-compare existing work and identify their limitations. Fortunately, several survey papers on sentiment analysis like [25, 37] helped me to speed up the process.

Pilot Work As a pilot study of my PhD, I have worked towards achieving the objective of alleviating data sparsity for Twitter sentiment analysis. This piece of my work is presented in more details in Section 3.1. The results have already been published in [31]. An extended study has been submitted to WWW conference [32].

It is worth mentioning that working on the problem of data sparsity had a great impact on the outcome and the progress of my work. It helped to develop my experience in the area of microblog sentiment analysis and increased my awareness of the challenges in this area. Moreover it drew the road map of my research by conducting more work which includes the following:

- A statistical study on political tweets data. The main aim of the study was to understand the relation between the public opinions on Twitter and the outcome of a certain political event. This study has been summarized in an abstract submitted to LREC [14].
- *Tweenator*,² an easy-to-use polarity annotation tool for tweets data. I built this tool in order to collect manually annotated tweets by crowd sourcing. Using this tool I succeeded in collecting 640 subjective tweets annotated by more than 10 different users. This corpus has been used as a testing dataset for our work in [32].
- *Moody* is a Facebook application for tracking and detecting friends' moods on Facebook. Two goals were behind this application. One is to try our current approach in [32] on Facebook status updates data. Another is to collect large amount of status updates data which can be used later for different research purposes. This work is still in progress.

Thus it can be shown that the first year of my research was dedicated to (1) develop my knowledge about the problems to be addressed, (2) understand existing work on microblog sentiment analysis, (3) formulate my research questions in answering to the challenges identified in the literature review, and finally (4) work simultaneously in different lines of research in order to build a coherent story of my research problems.

²<http://www.atkmi.com/tweenator/>

1.6 Report Overview

The rest of the report is organized as follows:

- Chapter 2 outlines existing work on sentiment analysis with focus on Twitter sentiment analysis.
- Chapter 3 presents my pilot work conducted during the first year including the proposed approach to alleviate the data sparsity for Twitter sentiment analysis and the study on political tweets data.
- Chapter 4 outlines the future plan for the remaining two years.

Chapter 2

Literature Survey

Sentiment analysis of conventional text is a relatively old problem. However, recent years have witnessed increased interests in sentiment analysis of microblogs. In this chapter I am going to present the literature of the previous work in sentiment analysis. In particular, I will focus more on recent work of sentiment analysis of microblogs, identify their problems, and provide a cross comparison at the end.

2.1 General Overview

Previous work on text-based sentiment analysis follows two main approaches: The first approach assumes that semantic orientation of a document is an averaged sum of the semantic orientations of its words and phrases. The pioneer work is the point-wise mutual information approach proposed in Turney [39]. Also work such as [13, 15, 34, 30] are good examples of this lexical-based approach. The second approach [26, 24, 6, 46, 22] addresses the problem as a text classification task where classifiers are built using one of the machine learning methods and trained on a dataset using features such as unigrams, bigrams, part-of-speech (POS) tags, etc. The vast majority of work in sentiment analysis mainly focuses on the domains of movie reviews, product reviews and blogs.

Although most of the work [26, 24, 6, 7] achieved relatively high sentiment classification accuracies, they suffered from the domain-dependence problem where performance often drops precipitously by applying the same classifiers on other domains of interest. Other work tried to overcome this problem by either building a hybrid classifier [18], or focusing on domain-independent features [43].

Works in [19, 21] have addressed the domain-independence problem by building weakly supervised classifiers where supervision comes from word polarity priors rather than labelled documents. They can extract the latent topics in a document with its associated sentiments.

Apart from sentiment analysis at the document level there have also been work on detecting sentiment at the phrase or sentence level [45, 42, 36].

2.2 Twitter Sentiment Analysis

Detecting sentiment from tweet data is considered as a much harder problem than sentiment analysis on conventional text such as review documents, mainly due to the short length of tweet

messages, the frequent use of informal and irregular words, the rapid evolution of language in Twitter, and the data streaming paradigm that Twitter has. Annotated tweets data are impractical to obtain. A large amount of work have been conducted on twitter sentiment analysis using noisy labels (also called distant supervision). For example, Go et al. [11] used emoticons such as “:-)” and “:(” to label tweets as positive or negative and train standard classifiers such as Naïve Bayes (NB), Maximum Entropy (MaxEnt), and Support Vector Machines (SVMs) to detect the sentiments of tweets. The best result of 83% was reported by MaxEnt using a combination of unigrams and bigrams. Barbosa and Feng [3] collected their training data from three different Twitter sentiment detection websites which mainly use some pre-built sentiment lexicons to label each tweet as positive or negative. Using SVMs trained from these noisy labeled data, they obtained 81.3% in sentiment classification accuracy.

While the aforementioned approaches did not detect neutral sentiment, Pak and Paroubek [23] additionally collected neutral tweets from Twitter accounts of various newspapers and magazines and trained a three-class NB classifier which is able to detect neutral tweets in addition to positive and negative tweets. Their NB was trained with a combination of n -grams and POS features.

Speriosu et al. [33] argued that using noisy sentiment labels may hinder the performance of sentiment classifiers. They proposed exploiting the Twitter follower graph to improve sentiment classification and constructed a graph that has users, tweets, word unigrams, word bigrams, hashtags, and emoticons as its nodes which are connected based on the link existence among them (e.g., users are connected to tweets they created; tweets are connected to word unigrams that they contain etc.). They then applied a label propagation method where sentiment labels were propagated from a small set of nodes seeded with some initial label information throughout the graph. They claimed that their label propagation method outperforms MaxEnt trained from noisy labels and obtained an accuracy of 84.7% on the subset of the twitter sentiment test set from [11].

There have also been some work in exploring feature engineering to improve the performance of sentiment classification on tweets. Agarwal et al. [1] studied using the feature based model and the tree kernel based model for sentiment classification. They explored a total of 50 different feature types and showed that both the feature based and tree kernel based models perform similarly and they outperform the unigram baseline.

Kouloumpis et al. [17] compared various features including n -gram features, lexicon features based on the existence of polarity words from the MPQA subjectivity lexicon¹, POS features, and microblogging features capturing the presence of emoticons, abbreviations, and intensifiers. They found that microblogging features are most useful in sentiment classification.

It can be observed that the aforementioned work on Twitter sentiment analysis follow the statistical classification approach where sentiment models do not change after being built. However, Twitter follow the data streaming paradigm. This means that huge amounts of data are transferred, viewed then discarded immediately. Therefore, sentiment classifiers have to deal with two new constraints, (1) They need to operate in a limited time and space environment and (2) deal with dramatic changes in sentimental data over time.

Bifet and Frank [5] proposed using Stochastic Gradient Decent (SGD) method for Twitter data stream sentiment analysis. They pointed out that class distribution may vary over time. For example, more positive tweets can be streamed than negative ones. This poses unbalanced classes problem for classifiers training. To address this problem they proposed using Kappa

¹<http://www.cs.pitt.edu/mpqa/>

statistic as measure for evaluating predictive accuracy of streaming classifiers. testing their approach on two different datasets, Twitter sentiment dataset from [11] and the Edinburgh corpus [28], they argued that Kappa performs better with unbalanced data than other measurements methods.

Recently, there has been some work on political sentiment analysis from microblogs. It has been shown that public sentiment from tweets data can be used as a good indicator of political preferences. Early work that investigates the political sentiment in microblogs was done by Tumasjan et al. [38] in which they analysed 104,003 tweets published in the weeks leading up to German federal election to predict election results. Tweets published over the relevant timeframe were concatenated into one text sample and are mapped into 12 emotional dimensions using the LIWC (Linguistic Inquiry and Word Count) software [27]. They found that the number of tweets mentioning a particular party is almost as accurate as traditional election polls which reflects the election results.

Diakopoulos and Shamma [9] tracked real-time sentiment pulse from aggregated tweet messages during the first U.S. presidential TV debate in 2008 and revealed affective patterns in public opinion around such a media event. Tweet message sentiment ratings were acquired using Amazon Mechanical Turk.

































Conover et al. [8] examined the retweet network, where users are connected if one re-tweet tweets produced by another, and the mention network, where users are connected if one has mentioned another in a tweet, of 250,000 political tweets during the six weeks prior to the 2010 U.S. midterm elections. They found that the retweet network exhibits a highly modular structure, with users being separated into two communities corresponding to political left and right. But the mention network does not exhibit such political segregation.

Livne et al. [20] studied the use of Twitter by almost 700 political party candidates during the midterm 2010 elections in the U.S. For each candidate, they performed structure analysis on the network constructed by the “*following*” relations; and content analysis on the user profile built using a language modeling (LM) approach. Logistic regression models were then built using a mixture of structure and content variables for election results prediction. They also found that applying LDA to the corpus failed to extract high-quality topics.

2.3 Discussion

It can be observed that the previous work mainly tackles two common challenges of (1) the lack of labelled data and (2) the shift in the studied domains. Other work addresses the problem of Twitter sentiment analysis from two different perspectives: (1) Feature engineering perspective. Work along this line argues that selecting better features leads to overcome both the problems of short length of tweet messages and language variations in Twitter. They essentially tried to overcome the data sparsity problem. (2) Dealing with streaming data. Work along this line emphasises that data should be processed in real time, which means classifiers should be able to work with limited resources of time and space, handle the unbalanced distribution of sentiments, and adapt the underlying models dynamically.

Figure 2.1 shows that the majority of the previous work has not tackled or partially tackled each of our major 3D challenges. Hence, sentiment analysis of microblogs is still a fresh and promising research area to be explored.

	Data Sparsity	Domain	Dynamics
Go et al. [9]			
Barbosa and Feng [2]			
Pak and Paroubek [20]			
Speriosu et al. [29]			
Agarwal et al. [1]			
Kouloumpis et al. [14]			
Bifet and Frank [4]			
Tumasjan et al. [33]			
Diakopoulos and Shamma [8]			
Conover et al. [7]			
Livne et al. [17]			



Tackled



Partially Tackled



Not Tackled

Figure 2.1: The coverage of previous work regarding about the 3D problems

Chapter 3

Pilot Work

3.1 Alleviating Data Sparsity for Twitter Sentiment Analysis

I have discussed in Chapter 2 that previous work on Twitter sentiment analysis [11, 23, 3] rely on noisy labels or distant supervision, for example, by taking emoticons as the indication of tweet sentiment, to train supervised classifiers. Other work explore feature engineering in combination of machine learning methods to improve sentiment classification accuracy on tweets [1, 17]. However, none of the work explicitly addressed the data sparsity problem which is one of the major challenges facing when dealing with tweets data.

Figure 3.1 shows the word frequency histogram from the tweets data we used for our experiments. Here, y -axis is in log-scale. It can be observed that out of a total of 96,084 distinct words, 74,185 words only occur less than 5 times. The total number of words occurring more than 10 times is merely 6,835.

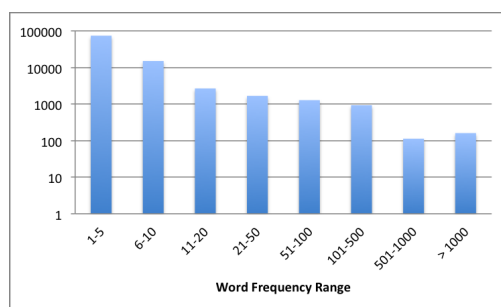


Figure 3.1: Word frequency histogram.

One possible way to alleviate data sparseness is through word clustering such that words contributing similarly to sentiment classification are grouped together. We propose two approaches to realise word clustering, one is through semantic smoothing [31], the other is through automatic sentiment-topics extraction. Semantic smoothing extracts semantically hidden concepts from tweets and then incorporates them into supervised classifier training by interpolation. An inspiring example for using semantic smoothing is shown in Figure 3.2 where the left box lists entities appeared in the training set together with their occurrence probabilities in positive and negative tweets. For example, the entities “*iPad*”, “*iPod*” and “*Mac Book Pro*” appeared more often in tweets of positive polarity and they are all mapped to the semantic concept “*Product/Apple*”. As a result, the tweet from the test set “*Finally, I got my iPhone. What a product!*”

is more likely to have a positive polarity because it contains the entity “*iPhone*” which is also mapped to the concept “*Product/Apple*”.

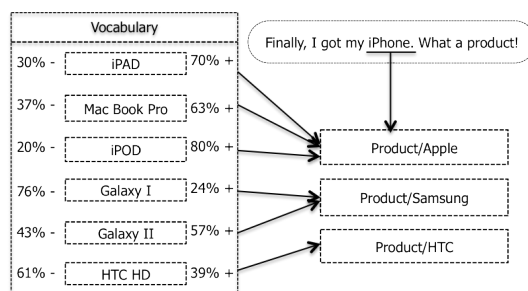


Figure 3.2: Incorporating semantic concepts for sentiment classification.

We propose a semantic interpolation method to incorporate semantic concepts into sentiment classifier training where we interpolate the original unigram language model in the Naïve Bayes (NB) classifier with the generative model of words given semantic concepts. We show on the Stanford Twitter Sentiment Data [11] that simply replacing words with their corresponding semantic concepts reduces the vocabulary size by nearly 20%. However, the sentiment classification accuracy drops by 4% compared to the baseline NB model trained on unigrams solely. With the interpolation method, the sentiment classification accuracy improves upon the baseline model by nearly 4%.

Our second approach for automatic word clustering is through sentiment-topics extraction using the previously proposed joint sentiment-topic (JST) model [19]. The JST model extracts latent topics and the associated topic sentiment from the tweets data which are subsequently added into the original feature space for supervised classifier training. Our experimental results show that NB learned from these features outperforms the baseline model trained on unigrams only and achieves the state-of-the-art result on the original test set of the Stanford Twitter Sentiment Data.

3.1.1 Twitter Sentiment Corpus

In this work, we used the Stanford Twitter Sentiment Data¹ which was collected between the 6th of April and the 25th of June 2009 [11]. The training set consists of 1.6 million tweets with the same number of positive and negative tweets labeled using emoticons. For example, a tweet is labeled as positive if it contains :), :-), :), :D, or =) and is labeled as negative if it has :(, :-(, or : (, etc. The original test set consists of 177 negative and 182 positive manually annotated tweets.

We built our training set by randomly selecting 60,000 balanced tweets from the original training set in the Stanford Twitter Sentiment Data. Since the original test set only contains a total of 359 tweets which is relatively small, we manually annotated additional 641 tweets from the original remaining training data. Our final test set contains 1,000 tweet messages with 527 negative and 473 positive tweets.

¹<http://twittersentiment.appspot.com/>

3.1.2 Semantic Features

Twitter is an open social environment where there are no restrictions on what users can tweet about. Therefore, a huge number of infrequent named entities, such as people, organization, products, etc., can be found in tweet messages. These infrequent entities make the data very sparse and hence hinder the sentiment classification performance. Nevertheless, many of these named entities are semantically related. For example, the entities “*iPad*” and “*iPhone*” can be mapped to the same semantic concept “*Product/Apple*”. Inspired by this observation, we propose using semantic features to alleviate the sparsity problem from tweets data. We first extract named entities from tweets and map them to their corresponding semantic concepts. We then incorporate these semantic concepts into NB classifier training.

Semantic Concept Extraction

We investigated three third-party services to extract entities from tweets data, Zemanta,² OpenCalais,³ and AlchemyAPI.⁴ A quick and manual comparison of a randomly selected 100 tweet messages with the extracted entities and their corresponding semantic concepts showed that AlchemyAPI performs better than the others in terms of the quality and the quantity of the extracted entities. Hence, we used AlchemyAPI for the extraction of semantic concepts in our paper.

Using AlchemyAPI, we extracted a total of 15,139 entities from the training set, which are mapped to 30 distinct concepts and extracted 329 entities from the test set, which are mapped to 18 distinct concepts. Table 3.1 shows the top five extracted concepts from the training data with the number of entities associated with them.

Concept	Number of Entities
Person	4954
Company	2815
City	1575
Country	961
Organisation	614

Table 3.1: Top 5 concepts with the number of their associated entities.

Incorporating Semantic Concepts into NB Training

The extracted semantic concepts can be incorporated into sentiment classifier training in a naive way where entities are simply replaced by their mapped semantic concepts in the tweets data. For example, all the entities such as “*iPhone*”, “*iPad*”, and “*iPod*” are replaced by the semantic concept “*Product/Apple*”. A more principled way to incorporate semantic concepts is through interpolation. Here, we propose interpolating the unigram language model with the generative model of words given semantic concepts in NB training.

²<http://www.zemanta.com/>

³<http://www.opencalais.com/>

⁴<http://www.alchemyapi.com/>

In NB, the assignment of a sentiment class c to a given tweet \mathbf{w} can be computed as:

$$\begin{aligned}\hat{c} &= \arg \max_{c \in \mathcal{C}} P(c|\mathbf{w}) \\ &= \arg \max_{c \in \mathcal{C}} P(c) \prod_{1 \leq i \leq N_{\mathbf{w}}} P(w_i|c),\end{aligned}\quad (3.1)$$

where $N_{\mathbf{w}}$ is the total number of words in tweet \mathbf{w} , $P(c)$ is the prior probability of a tweet appearing in class c , $P(w_i|c)$ is the conditional probability of word w_i occurring in a tweet of class c .

In multinomial NB, $P(c)$ can be estimated by $P(c) = N_c/N$ Where N_c is the number of tweets in class c and N is the total number of tweets. $P(w_i|c)$ can be estimated using maximum likelihood with Laplace smoothing:

$$P(w|c) = \frac{N(w, c) + 1}{\sum_{w' \in V} N(w'|c) + |V|} \quad (3.2)$$

Where $N(w, c)$ is the occurrence frequency of word w in all training tweets of class c and $|V|$ is the number of words in the vocabulary. Although using Laplace smoothing helps to prevent zero probabilities of the “unseen” words, it assigns equal prior probabilities to all of these words.

We propose a new smoothing method where we interpolate the unigram language model in NB with the generative model of words given semantic concepts. Thus, the new class model with semantic smoothing has the following formula:

$$P_s(w|c) = (1 - \alpha)P_u(w|c) + \alpha \sum_j P(w|s_j)P(s_j|c)$$

Where $P_s(w|c)$ is the unigram class model with semantic smoothing, $P_u(w|c)$ is the unigram class model with maximum likelihood estimate, s_j is the j -th concept of the word w , $P(s_j|c)$ is the distribution of semantic concepts in training data of a given class and it can be computed via the maximum likelihood estimation. $P(w|s_j)$ is the distribution of words in the training data given a concept and it can be also computed via the maximum likelihood estimation. Finally, the coefficient α is used to control the influence of the semantic mapping in the new class model. By setting α to 0 the class model becomes a unigram language model without any semantic interpolation. On the other hand, setting α to 1 reduces the class model to a semantic mapping model. In this work, α was empirically set to 0.5.

3.1.3 Sentiment-Topic Features

The joint sentiment-topic (JST) model [19] is a four-layer generative model which allows the detection of both sentiment and topic simultaneously from text. The generative procedure under JST boils down to three stages. First, one chooses a sentiment label l from the per-document sentiment distribution π_d . Following that, one chooses a topic z from the topic distribution $\theta_{d,l}$, where $\theta_{d,l}$ is conditioned on the sampled sentiment label l . Finally, one draws a word w_i from the per-corpus word distribution $\phi_{l,z}$ conditioned on both topic z and sentiment label l . The JST model does not require labelled documents for training. The only supervision is word prior polarity information which can be obtained from publicly available sentiment lexicons such as the MPQA subjectivity lexicon.

We train JST on the combined training and test sets with tweet sentiment labels being discarded. The resulting model assigns each word in tweets with a sentiment label and a topic label. Hence, JST essentially clusters different words sharing similar sentiment and topic. We list some of the topic words extracted by JST in Table 3.2. Words in each cell are grouped under one topic and the upper half of the table shows topic words bearing positive sentiment while the lower half shows topic words bearing negative polarity. It can be observed that words groups under different sentiment and topic are quite informative and coherent. For example, Topic 3 under positive sentiment is related to a good music album, while Topic 1 under negative sentiment is about a complaint of feeling sick possibly due to cold and headache.

	Topic 1	Topic 2	Topic 3	Topic 4	Topic 5
Positive	dream	bought	song	eat	movi
	sweet	short	listen	food	show
	train	hair	love	coffe	award
	angel	love	music	dinner	live
	love	wear	play	drink	night
	goodnight	shirt	album	yummi	mtv
	free	dress	band	chicken	concert
	club	photo	guitar	tea	vote
Negative	feel	miss	rain	exam	job
	today	sad	bike	school	hard
	hate	cry	car	week	find
	sick	girl	stop	tomorrow	hate
	cold	gonna	ride	luck	interview
	suck	talk	hit	suck	lost
	weather	bore	drive	final	kick
	headache	feel	run	studi	problem

Table 3.2: Extracted polarity words by JST.

Inspired by the above observations, grouping words under the same topic and bearing similar sentiment could potentially reduce data sparseness in twitter sentiment classification. Hence, we extract sentiment-topics from tweets data and augment them as additional features into the original feature space for NB training. Algorithm 1 shows how to perform NB training with sentiment-topics extracted from JST. The training set consists of labeled tweets, $\mathcal{D}^{train} = \{(\mathbf{w}_n; c_n) \in \mathcal{W} \times \mathcal{C} : 1 \leq n \leq N^{train}\}$, where \mathcal{W} is the input space and \mathcal{C} is a finite set of class labels. The test set contains tweets without labels, $\mathcal{D}^{test} = \{\mathbf{w}_n^t \in \mathcal{W} : 1 \leq n \leq N^{test}\}$. The training and test sets are first merged with tweets sentiment labels discarded. A JST model is then learned from the merged corpus to generate sentiment-topics for each tweet. The original tweets are augmented with those sentiment-topics as shown in Step 4 of Algorithm 1, where l_i-z_i denotes a combination of sentiment label l_i and topic z_i for word w_i . Finally, an optional feature selection step can be performed according to the information gain criteria and a classifier is then trained from the training set with the new feature representation.

Algorithm 1 NB training with sentiment-topics extracted from JST.

Input: The training set \mathcal{D}^{train} and test set \mathcal{D}^{test}

Output: NB sentiment classifier

- 1: Merge \mathcal{D}^{train} and \mathcal{D}^{test} with document labels discarded, denote the merged set as \mathcal{D}
 - 2: Train a JST model on \mathcal{D}
 - 3: **for** each tweet $\mathbf{w}_n = (w_1, w_2, \dots, w_m) \in \mathcal{D}$ **do**
 - 4: Augment tweet with sentiment-topics generated from JST,
 $\mathbf{w}'_n = (w_1, w_2, \dots, w_m, l_{1-z_1}, l_{2-z_2}, \dots, l_{m-z_m})$
 - 5: **end for**
 - 6: Create a new training set $\mathcal{D}^{train'} = \{(\mathbf{w}'_n; c_n) : 1 \leq n \leq N^{train}\}$
 - 7: Create a new test set $\mathcal{D}^{test'} = \{\mathbf{w}'_n : 1 \leq n \leq N^{test}\}$
 - 8: Perform feature selection using IG on $\mathcal{D}^{train'}$
 - 9: Return NB trained on $\mathcal{D}^{train'}$
-

Pre-processing	Vocabulary Size	% of Reduction
None	95,130	0%
Username	70,804	25.58%
Hashtag	94,200	0.8%
URLS	92,363	2.91%
Repeated Letters	91,824	3.48%
Digits	92,785	2.47%
Symbols	37,054	29.47%
All	37,054	61.05%

Table 3.3: The effect of pre-processing.

3.1.4 Experimental Results

In this section, we present the results obtained on the twitter sentiment data using both semantic features and sentiment-topic features and compare with the existing approaches.

Pre-processing

The raw tweets data are very noisy. There are a large number of irregular words and non-English characters. Tweets data have some unique characteristics which can be used to reduce the feature space through the following pre-processing:

- All Twitter usernames, which start with @ symbol, are replaced with the term “USER”.
- All URL links in the corpus are replaced with the term “URL”.
- Reduce the number of letters that are repeated more than twice in all words. For example the word “loooooveeee” becomes “loovee” after reduction.
- Remove all Twitter hashtags which start with the # symbol, all single characters and digits, and non-alphanumeric characters.

Table 3.3 shows the effect of pre-processing on reducing features from the original feature space. After all the pre-processing, the vocabulary size is reduced by 62%.

Semantic Features

We have tested both the NB classifier from WEKA⁵ and the maximum entropy (MaxEnt) model from MALLET⁶. Our results show that NB consistently outperforms MaxEnt. Hence, we use NB as our baseline model. Table 3.4 shows that with NB trained from unigrams only, the sentiment classification accuracy of 80.7% was obtained.

We extracted semantic concepts from tweets data using Alchemy API and then incorporated them into NB training by the following two simple ways. One is to replace all entities in the tweets corpus with their corresponding semantic concepts (*semantic replacement*). Another is to augment the original feature space with semantic concepts as additional features for NB training (*semantic augmentation*). With *semantic replacement*, the feature space shrunk substantially by nearly 20%. However, sentiment classification accuracy drops by 4% compared to the baseline as shown in Table 3.4. The performance degradation can be explained as the mere use of semantic concepts replacement which leads to information loss and subsequently hurts NB performance. Augmenting the original feature space with semantic concepts performs slightly better than *semantic replacement*, though it still performs worse than the baseline.

With *Semantic interpolation*, semantic concepts were incorporated into NB training taking into account the generative probability of words given concepts. The method improves upon the baseline model and gives a sentiment classification accuracy of 84%.

Method	Accuracy
Unigrams	80.7%
Semantic replacement	76.3%
Semantic augmentation	77.6%
Semantic interpolation	84.0%
Sentiment-topic features	82.3%

Table 3.4: Sentiment classification results on the 1000-tweet test set.

Sentiment-Topic Features

To run JST on the tweets data, the only parameter we need to set is the number of topics T . It is worth noting that the total number of the sentiment-topics that will be extracted is $3 \times T$. For example, when T is set to 50, there are 50 topics under each of positive, negative and neutral sentiment labels. Hence the total number of sentiment-topic features is 150. We augment the original bag-of-words representation of the tweet messages by the extracted sentiment-topics. Figure 3.3 shows the classification accuracy of NB trained from the augmented features by varying the number of topics from 1 to 65. The initial sentiment classification accuracy is 81.1% with topic number 1. Increasing the number of topics leads to the increase of classification accuracy with the peak value of 82.3% being reached at topic number 50. Further increasing topic numbers degrades the classifier performance.

⁵<http://www.cs.waikato.ac.nz/ml/weka/>

⁶<http://mallet.cs.umass.edu/>

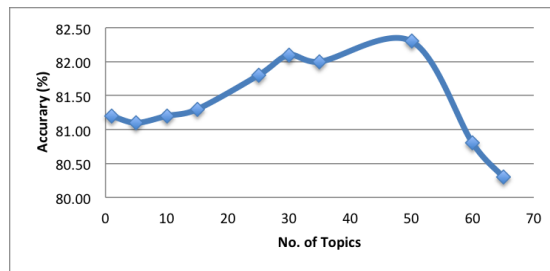


Figure 3.3: Classification accuracy vs. number of topics.

Comparison with Existing Approaches

In order to compare our proposed methods with the existing approaches, we also conducted experiments on the original Stanford Twitter Sentiment test set which consists of 177 negative and 182 positive tweets. The results are shown in Table 3.5. The sentiment classification accuracy of 83% reported in [11] was obtained using MaxEnt trained on a combination of unigrams and bigrams. It should be noted that while Go et al. used 1.6 million tweets for training, we only used a subset of 60,000 tweets as our training set.

Speriosu et al. [33] tested on a subset of the Stanford Twitter Sentiment test set with 75 negative and 108 positive tweets. They reported the best accuracy of 84.7% using label propagation on a rather complicated graph that has users, tweets, word unigrams, word bigrams, hashtags, and emoticons as its nodes.

It can be seen from Table 3.5 that *sentiment replacement* performs worse than the baseline. *Sentiment augmentation* does not result in the significant decrease of the classification accuracy, though it does not lead to the improved performance either. Our *semantic interpolation* method rivals the best result reported on the Stanford Twitter Sentiment test set. Using the sentiment-topic features, we achieved 86.3% sentiment classification accuracy, which outperforms the existing approaches.

Method	Accuracy
Unigrams	81.0%
Semantic replacement	77.3%
Semantic augmentation	80.45%
Semantic interpolation	84.1%
Sentiment-topic features	86.3%
(Go et al., 2009)	83%
(Speriosu et al., 2011)	84.7%

Table 3.5: Sentiment classification results on the original Stanford Twitter Sentiment test set.

Discussion

We have explored incorporating semantic features and sentiment-topic features for twitter sentiment classification. While simple *semantic replacement* or *augmentation* does not lead to the improvement of sentiment classification performance, *sentiment interpolation* improves upon the baseline NB model trained on unigrams only by 3%. Augmenting feature space with sentiment-topics generated from JST also results in the increase of sentiment classification

accuracy compared to the baseline. On the original Stanford Twitter Sentiment test set, NB classifiers learned from sentiment-topic features outperform the existing approaches.

We have a somewhat contradictory observation here. Using sentiment-topic features performs worse than using semantic features on the test set comprising of 1000 tweets. But the reverse is observed on the original Stanford Twitter Sentiment test set with 359 tweets. We therefore conducted further experiments to compare these two approaches.

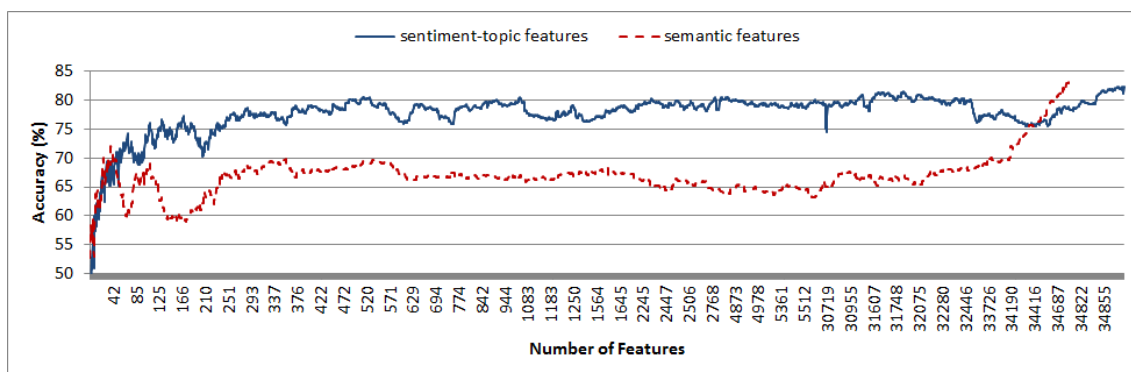


Figure 3.4: Classification accuracy vs. number of features selected by information gain.

We performed feature selection using information gain (IG) on the training set. We calculated the IG value for each feature and sorted them in descending order based on IG. Using each distinct IG value as a threshold, we ended up with different sets of features to train a classifier. Figure 3.4 shows the sentiment classification accuracy on the 1000-tweet test set versus different number of features. It can be observed that there is an abrupt change in x -axis from around 5600 features jumping to over 30,000 features. Using sentiment-topic features consistently performs better than using semantic features. With as few as 500 features, augmenting the original feature space with sentiment-topics already achieves 80.2% accuracy. Although with all the features included, NB trained with semantic features performs better than that with sentiment-topic features, we can still draw a conclusion that sentiment-topic features should be preferred over semantic features for the sentiment classification task since it gives much better results with far less features.

3.2 Quantising Opinions for Political Tweets Analysis

There have been increasing interests in recent years in analyzing tweet messages relevant to political events so as to understand public opinions towards certain political issues. We analyzed tweet messages crawled during the eight weeks leading to the UK General Election in May 2010 and found that activities at Twitter is a good predictor of popularity of political parties. We also proposed a statistical model for sentiment detection with side information such as emoticons and hash tags implying tweet polarities being incorporated. Our results show that sentiment analysis based on a simple keyword matching against a sentiment lexicon does not correlate well with the actual election results. However, using our proposed statistical model for sentiment analysis, we were able to map the public opinion in Twitter with the actual offline sentiment in real world.

3.2.1 Objective

As shown in Section 3.1, JST model has been used to extract sentiment-topic features from the training set. Later on, These features were incorporated as additional features into sentiment classifier training. An urgent question one could ask here is: how accurate are the extracted features. Experimental results on movie review and products review data as reported in [19] show that JST model outperforms other existing semi-supervised approaches. However, it is still unknown how JST model performs on microblogging data. One problem facing us when evaluating JST model on the data of microblogs is the lack of ground truth data this is needed for the evaluation. To overcome this problem we run JST on political tweets data of the UK General Election in May 2010 and compared the results with the actual election results. The experimental results, as shown in this section, gave us a relatively good estimation of how JST model perform on tweets data.

3.2.2 Political Tweet Corpus

The tweets data we used in this work were collected using the Twitter Streaming API⁷ for 8 weeks leading to the UK general election in 2010. Search criteria specified include the mention of political parties such as Labour, Conservative, Tory, etc.; the mention of candidates such as Brown, Cameron, Clegg, etc.; the use of the hash tags such as #election2010, #Labour etc.; and the use of certain words such as “election”. After removing duplicate tweets in the downloaded data, the final corpus contains 919,662 tweets.

There are three main parties in the UK General Election 2010, Conservative, Labour, and Liberal Democrat. We first categorized tweet messages as in relevance to different parties by comparing keywords and hashtags against a manually constructed list. Figure 3.5 show tweets volume distributions for different parties. It can be observed that among the three main UK parties, Conservative appears to be the most popular one with over 41% relevant tweets. The Labour Party only attracted 9% tweets, indicating that Twitter users have lost interests in Labour.

Table 3.6 lists the top 10 most influential users ranked by the number of followers, the number of re-tweets, and the number of mentions. The ranked list by the number of followers contains mostly news media organisations and it does not overlap much with the other two ranked lists. On the contrary, the ranked lists by the number of re-tweets and the number of mentions have 6 users in common. We also notice that although the total tweets volume for

⁷<https://dev.twitter.com/docs/streaming-api>

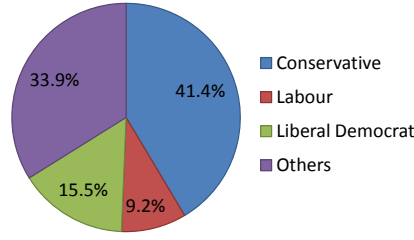


Figure 3.5: The tweets volume distribution for various parties.

the Labour Party appears to be the least as shown in Figure 3.5, it has the most number of both re-tweets and mentions.

Table 3.6: Social influence ranking results.

<i>No. of Followers</i>	<i>No. of Re-tweets</i>	<i>No. of Mentions</i>
CNN Breaking News	UK Labour Party	UK Labour Party
The New York Times	Ben Goldacre	John Prescott
Perez Hilton	Eddie Izzard	Eddie Izzard
Good Morning America	John Prescott	Ben Goldacre
Breaking News	Sky News	The Labour Party
Women's Wear Daily	David Schneider	Conservatives
Guardian Tech	Conservatives	Ellie Gellard
The Moment	Mark Curry	Alastair Campbell
Eddie Izzard	Stephen Fry	Sky News
Ana Marie Cox	Carl Maxim	Nick Clegg

3.2.3 Tweet Sentiment Analysis

We have previously proposed the joint sentiment-topic (JST) model which is able to extract polarity-bearing topics from text and infer document-level polarity labels. The only supervision required is a set of words marked with their prior polarity information.

Assume that we have a corpus with a collection of D documents denoted by $C = \{d_1, d_2, \dots, d_D\}$; each document in the corpus is a sequence of N_d words denoted by $d = (w_1, w_2, \dots, w_{N_d})$, and each word in the document is an item from a vocabulary index with V distinct terms denoted by $\{1, 2, \dots, V\}$. Also, let S be the number of distinct sentiment labels, and T be the total number of topics. The generative process in JST which corresponds to the graphical model shown in Figure 3.6(a) is as follows:

- For each document d , choose a distribution $\pi_d \sim \text{Dir}(\gamma)$.
- For each sentiment label l under document d , choose a distribution $\theta_{d,l} \sim \text{Dir}(\alpha)$.
- For each word w_i in document d
 - choose a sentiment label $l_i \sim \text{Mult}(\pi_d)$,
 - choose a topic $z_i \sim \text{Mult}(\theta_{d,l_i})$,

- choose a word w_i from $\varphi_{z_i}^{l_i}$, a Multinomial distribution over words conditioned on topic z_i and sentiment label l_i .



Figure 3.6: The Joint Sentiment-Topic (JST) model and the modified JST with side information incorporated.

Although the appearance of emoticons is not significant in our political tweets corpus, adding such side information has potential to further improve the sentiment detection accuracy. In the political tweets corpus, we also noticed that apart from emoticons, the used of hashtags could indicate polarity or emotion of the tweets. For example, the hashtag “#torywin” might represent a positive feeling towards the Tory (Conservative) Party, while “#labourout” could imply a negative feeling about the Labour Party. Hence, it would be useful to gather such side information and incorporate it into JST learning.

We show in the modified JST model in Figure 3.6(b) that the side information such as emoticons or hashtags indicating the overall polarity of tweets can be incorporated by updating the Dirichlet prior, γ , of the document-level sentiment distribution. In the original JST model, γ is a uniform prior and is set as $\gamma = (0.05 \times L)/S$, where L is the average document length, and the value of 0.05 on average allocates 5% of probability mass for mixing. In our modified model here, a transformation matrix η of size $D \times S$ is used to capture the side information as soft constraints. Initially, each element of η is set to 1. If the side information of a document d is available, then its corresponding elements in η is updated as:

$$\eta_{ds} = \begin{cases} 0.9 & \text{For the inferred sentiment label} \\ 0.1/(S-1) & \text{otherwise} \end{cases}, \quad (3.3)$$

where S is the total number of sentiment labels. For example, if a tweet contains “:-)”, then it is very likely that the sentiment label of the tweet is positive. Here, we set the probability of a tweet being positive to 0.9. The remaining 0.1 probability is equally distributed among the remaining sentiment labels. We then modify the Dirichlet prior γ by multiplying with the transformation matrix η .

We implemented a baseline model which simply assigns a score +1 and -1 to any matched positive and negative word respectively based on a sentiment lexicon. A tweet is then classified as either positive, negative, or neutral according to the aggregated sentiment score. We used the MPQA sentiment lexicon⁸ for both the baseline model labeling and for providing prior word polarity knowledge into the JST model learning.

⁸<http://www.cs.pitt.edu/mpqa/>

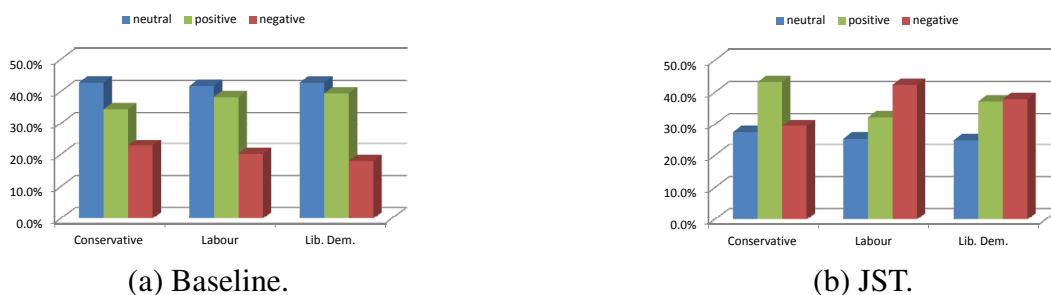


Figure 3.7: Sentiment distributions of the three main UK parties.

Figure 3.7 shows the sentiment distribution of the three main UK parties in tweets. It can be observed that the baseline model based on merely polarity word counts gives the similar trend on the three parties. The neutral tweets outnumbered positive tweets which in turn outnumbered negative tweets. However, the results from the JST model shows that the Conservative Party receives more positive marks where 43.2% express a favorable opinion. The Labour Party gets more negative views. The Liberal Democrat has roughly the same positive and negative views.

3.2.4 Discussion

In this work, we have analyzed tweet messages leading to the UK General Election 2010 to see whether they reflect the actual political landscape. Our results show that activities on the Twitter indeed indicate the popularity of election parties. Moreover, we have extended from our previously proposed joint sentiment-topic model by incorporating side information from tweets which include emoticons and hashtags that are indicative of polarities. The aggregated sentiment results are more closely match the offline public sentiment as compared to the simple lexicon-based approach.

3.3 Tweenator

Tweenator⁹ is a web based sentiment annotation tool for tweets Data. It allows users to easily assign a sentiment polarity to tweet messages, i.e. assign a negative, positive or neutral label to a certain tweet with regards to its contextual polarity. I used Tweenator to enlarge the sentiment corpus, which has been used in our work on data sparsity problem as discussed in Section 3.1.1 I was able to collect around 640 tweet messages within 6 days. 12 people have participated in the annotation process. They reported that they were able to annotate 10 tweet messages in 2 to 3 minutes approximately. Figure 3.8 shows the annotation user interface in Tweenator.

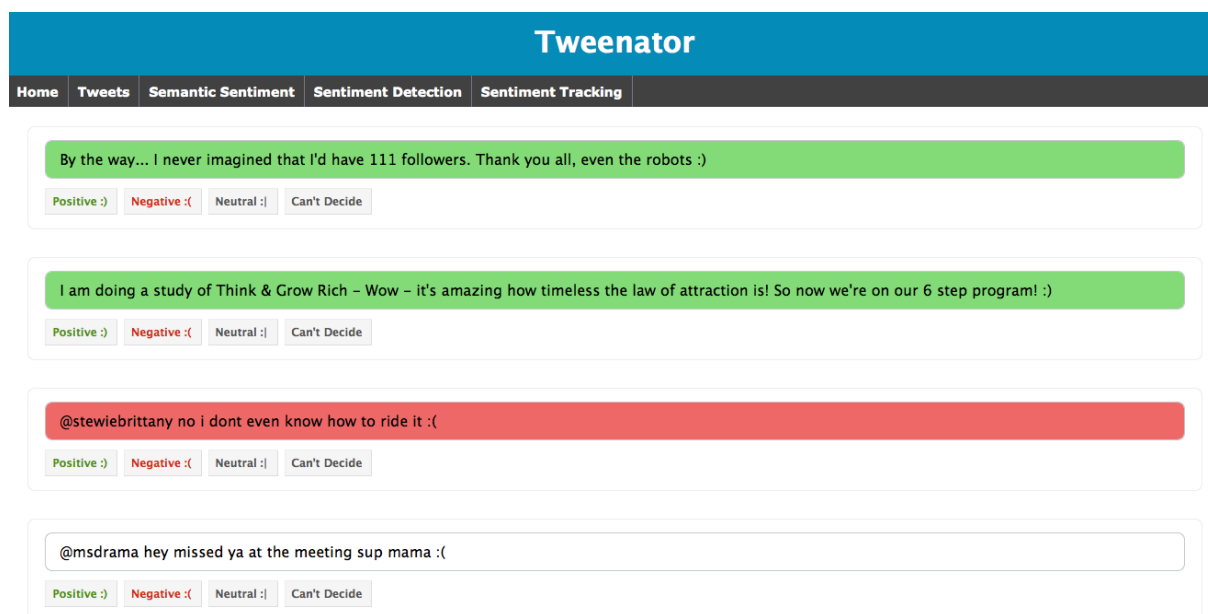


Figure 3.8: Tweenator as a sentiment annotation tool

Recently, I have implemented my approach for Twitter sentiment analysis using semantic features (See Section 3.1.2) as a new plug-in Tweenator. It provides the following features:

- Free-Form sentiment detector: this feature allows users to extract the polarity of their textual entry under the limit of 140 characters (See Figure 3.9-a).
- Opinionated tweet message retrieval tool. It allows users to extract negative/positive tweets towards a specific search term. For example a user can extract opinionated tweet message about the search term “Nike” (See Figure 3.9-b)

While the new plug-in is still under development, the trial alpha version is now available on our local server as shown in Figure 3.9.

⁹<http://atkmi.com/tweenator/>

The screenshot shows the Tweenator web application interface. At the top is a blue header with the word "Tweenator" in white. Below the header is a dark grey navigation bar with five menu items: "Home", "Tweets", "Semantic Sentiment", "Sentiment Detection", and "Sentiment Tracking". The "Semantic Sentiment" menu item is highlighted. Below the navigation bar is a search input field containing the text "I love my new iPhone" and a "Go" button. Below the search field are two horizontal bars representing sentiment results: a green bar with the text "I love my new iPhone" and a red bar with the text "I hate using ipad".

(a) Free-Form Sentiment Detector Interface.

The screenshot shows the Tweenator web application interface. At the top is a blue header with the word "Tweenator" in white. Below the header is a dark grey navigation bar with five menu items: "Home", "Tweets", "Semantic Sentiment", "Sentiment Detection", and "Sentiment Tracking". The "Sentiment Detection" menu item is highlighted. Below the navigation bar is a search input field containing the text "Nike" and a "Go" button. Below the search field are five horizontal bars representing tweet results with sentiment labels: a green bar with the text "@evelynbyrne have you tried Nike ? V. addictive.", a red bar with the text "Found NOTHING at Nike Factory :/ Off to Banana Republic Outlet! <http://myloc.me/2zic>", a red bar with the text "@Fraggle312 oh those are awesome! i so wish they weren't owned by nike :(", a green bar with the text "The Nike Training Club (beta) iPhone app looks very interesting.", and a green bar with the text "i love Dwight Howard's vitamin water commercial... now i wish he was with NIKE and not adidas. lol."

(b) Opinionated Tweet Message Retrieval Interface.

Figure 3.9: Sentiment Plug-in Interfaces in Tweenator

3.4 Moody - Sentiment Detection on Facebook

I have shown earlier that most of the work on sentiment analysis of microblogs was centered around Twitter. Very few work has been conducted on sentiment analysis on other microblogging sites such as Facebook due to the difficulties in collecting data and strict privacy issues. Facebook has some features that are different from Twitter, for example, (1) a status update can contains up to 420 characters, (2) Facebook distinguishes between a status updates, comments, links to external multimedia contents, and full articles, and (3) Status updates are usually connected to other social activities that users can take, for example, users can either “like” or comment on a status update.

Moody is a Facebook application that is able to extract sentiment from users’ status updates on Facebook. My work on Moody meets my final research objective of testing the proposed approaches in different microblogging environments (See Section 1.4).

3.4.1 How does Moody work?

Using the Open Graph protocol, which is provided by Facebook, Moody could be easily integrated into the Facebook social graph. Moody at this stage is still under development. However, the future vision that I have about Moody does work regarding the following scenario:

- Moody can access user’s status updates once the user authorizes the application
- According to Facebook privacy policy, Moody can retrieve user’s status updates for the last three month.
- Once Moody collect these data, it will be stored on our remote database on the server with all the related information.
- After that, the proposed methods will extract sentiment from this data and store the results on the server. Once a user log in to Moody, it will show him his status updates assigned with their sentiment labels.
- In case of mislabelling, the user can correct the sentiment label of a specific status update.

3.4.2 Two good things about Moody

Facebook has a very strict privacy policy regarding about user data. At the same time, it also has a fantastic authorization protocol. According to this protocol, once a user authorizes your application, the application can access the user’s data anytime. This actually provides two important advantages:

- The Offline Processing Mood: the application can access user’s data at any time, which means that the application can process the data at any time. Therefor, Moody does not always need to process the data in real time.
- Access Friends Data: Using the graph protocol with the right access permissions, Moody is not only able to access the status updates of a specific user, but also able to access

all status updates of all of his friends. In fact this is a great opportunity that Moody can use in: (1) Building the sentiment network of a certain user where its nodes are the sentiment orientations of the users within the network and the arcs are the social connections between those users. (2) Collecting a large amount of status updates data in a relatively short time.

Chapter 4

Future Work

Sentiment Analysis of microblogs faces many challenges, mainly due to the short length of status updates and the dynamic nature of the data. Previously in this study, I pointed out 5 objectives that should be achieved when working on sentiment analysis of microblogs. For the first objective, I have hypothesized that alleviating data sparsity problem leads to improve the performance of sentiment classifiers. To justify this hypothesis, two approaches were introduced to alleviate the data sparsity problem in Twitter sentiment classification. Experimental results show that our sentiment classifiers perform better than existing approaches. Nevertheless, much work still needs to be conducted to achieve the this objective. First, in the semantic method all entities were simply replaced by the associated semantic concepts. It is worth to perform a selective statistical replacement, which is determined based on the contribution of each concept towards making a better classification decision. Second, sentiment-topics generated by JST model were simply augmented into the original feature space of tweets data. It could lead to better performance by attaching a weight to each extracted sentiment-topic feature in order to control the impact of the newly added features. Finally, the performance of the NB classifiers learned from semantic features depends on the quality of the entity extraction process and the entity-concept mapping method. It is worth to investigate a filtering method, which can automatically filter out low-confidence semantic concepts.

It is easy to observe that the semantic model I proposed and the aforementioned future work that needs to be done to improve the model, are both trying to address the data sparsity problem by doing the analysis at the content level, in other words at the tweet level. One promising way to improve the proposed semantic model is to do the analysis at the user level where sentiment will be inferred based on users cultural background information. My hypothesis behind this is that:

“Users who have similar cultural backgrounds tend to have similar opinions about certain topic”

The following section gives more details about my future work of doing the sentiment at the user level or what is so called “Culture-Aware Sentiment Analysis”. This section will be followed by the progress plan the needs to be taken for the next two years.

4.1 Culture-Aware Sentiment Analysis

Microblogging platforms operate in an open environment where millions of users of different cultural backgrounds share their thoughts and opinions everyday. I propose a culture-aware model for dynamic sentiment analysis of large-scale streaming data of microblogs. The novelty of this model lies in the incorporation of cultural information of microblogging users into sentiment classifiers training. The proposed model works by grouping users into distinct clusters based on their cultural backgrounds, modelling each cluster as a set of cultural features, then incorporating these features into the sentiment classification process to alleviate the data sparsity problem and facilitate real-time sentiment analysis of large-scale streaming data.

4.1.1 Background

Many studies have shown that there is a strong correlation between users' cultural backgrounds and their attitudes and opinions towards certain topic or product. Assadi [2] showed that customers' preferences and tastes are dominated by their religious beliefs. Other studies [10, 40, 29] explore the impact of national culture on products adoption behaviours. They reported that there are many national factors that control the diffusion and the adoption rate of certain products in different countries.

It can be observed from the aforementioned studies that it is possible to predict users' attitudes or opinions towards certain product by modelling their cultural backgrounds as measurable factors. This observation motivates building a sentiment classification model that incorporates users' cultural information as additional features into sentiment classifiers training.

Very recent work [12, 35] has exploited static features of social networks by either using temporal information about users, or utilizing follower-followee social relations on Twitter to infer sentiment at the user level. While they focus on individual users at micro-level, I instead propose to extract common user cultural features from cultural groups at macro-level.

I propose a cultural-aware sentiment analysis model that use users' cultural information to improve sentiment prediction accuracy and facilitate real-time sentiment analysis.

4.1.2 Approach

Culture is an abstract concept that has many definitions. One definition suggests that culture is a set of shared attitudes, practices and behaviours that characterizes groups. This definition from one perspective can be interpreted as: online users, which have similar cultural backgrounds, might have similar opinions about a certain topic. For example, users who have British nationality and live in London are more likely to support the Conservative party in UK. From another perspective, this definition does also imply that users with similar opinions might have similar cultural background.

Inspired by the aforementioned intuition, the proposed model, as shown in Figure 4.1, can be built based on the following recursive process:

“Opinions define cultural groups and cultural groups infer opinions”

The proposed model works as follows:

1. Users are classified into distinct groups based on their opinions towards certain topic.

2. Similar profiling information of users in each group will be used to model the cultural features of the group.
3. Extracted cultural features will be used to update the underlying sentiment model.
4. Cultural groups will be fine-tuned using the updated sentiment model.

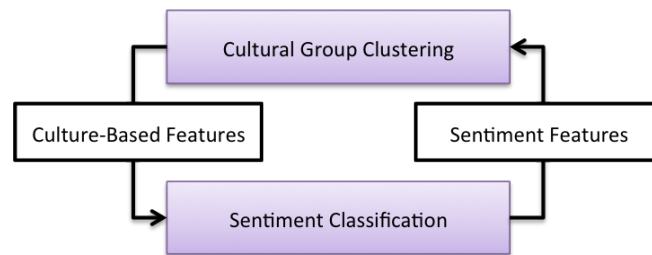


Figure 4.1: Culture-based sentiment model

By the end of the classification process we will have a set of users that are grouped together based on their cultural features.

From data analysis perspective, three different tasks will be conducted:

Culture-based feature analysis: General information about users such as religions, nationalities, geographic regions and racial groups can be used as static features to define different groups of users. However, cultural features are polymorphic and context-dependent. Different cultural feature sets will be used depending on the topic. For example, features like nationality and political views can be used to determine different cultural groups about the topic “UK Labour Party”. However, they are irrelevant in categorising cultural group on the topic “Yahoo Messenger”.

Incorporation of users’ cultural features into sentiment classification: I have previously proposed a semantic approach for Twitter sentiment analysis [31, 32]. This approach extracts semantically hidden concepts from tweets data and then incorporates them into supervised classifier training by interpolation. The interpolation method works by interpolating the original unigram language model in Nave Bays (NB) classifier with the generative model of words given semantic concept. Cultural features can be incorporated in a similar way where the unigram language model will also be interpolated by the generative model of users given cultural features.

Real-Time sentiment analysis: Microblogging services in general operate in a streaming fashion where data is transferred viewed and discarded immediately. This requires that sentiment classifiers should work in real time. Users’ opinions towards certain topic do not change suddenly. Hence, we can assume that their opinions remain unchanged over a certain period of time. Thus, instead of processing each individual tweet, we perform sentiment detection at macro-level or group-level where user’s opinion will be predicted based on his cultural group.

4.2 Progress Plan

In theory, I have 5 objectives to be achieved. However, each one of these objectives is considered as stand-alone problem in different research area. This make working on all of these objectives infeasible to achieve, especially within the limited period of the two years that I still have. Therefore, my plan is to focus on achieving Obj1 and Obj3 by conducting more work on incorporating user cultural features into my proposed semantic model as discussed in Section 4.1. The plan also includes other activities that are shown in Figure 4.2 and elaborated below:

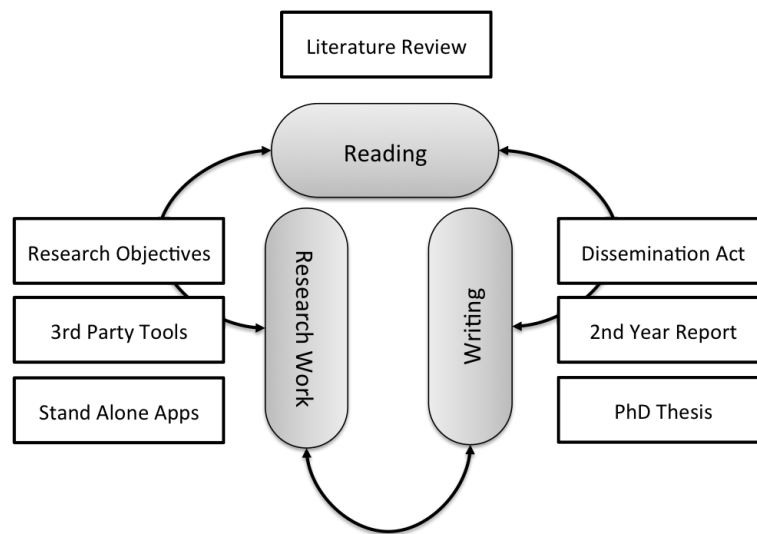


Figure 4.2: Main Tasks

1. Literature Review: It is a continuous process that started in the first year and will end by the end of the third year. However, for the next two years more attention will be paid to study the literature that is relevant to the objectives Obj1 and Obj3.
2. Research Work: It contains all research tasks that relate to my research plan including:
 - Conducting research work to achieve the research objectives (Obj1 & Obj3).
 - Building stand alone applications: Moody and Tweenator are good example of such applications, whose their main purpose is to test my approaches and mythologies in a large scale. These two applications will be continuously updated and extended as my research progresses during my PhD study.
3. Writing up: this includes the following writing activities:
 - Dissemination Activities: this will be conducted through publishing our work on main conferences and workshops.
 - Second year report (Technical Report)
 - Thesis Writing.

Table 4.1 summarizes the main tasks planned for the second and third year of this research work. It can be shown that efforts in the second year focus on developing methodologies that meet my research objectives. On the other hand, the third year will focus on dissemination work by conducting publishing activities throughout standard channels like main conferences and journals.

Second Year				
	Q1	Q2	Q3	Q4
1. Literature Review	X	X	X	X
2. Cultural-aware sentiment Analysis (Obj1)	X	X	X	X
2.1 Cultural Feature Analysis	X	X		
2.1.1 Data Collection	X			
2.1.2 Feature Extraction & Specification		X	X	
2.2 Incorporation of users' cultural features into sentiment classification			X	X
2.2.1 Interpolation into the proposed semantic model			X	X
2.2.3 Incorporate cultural features as prior information into JST model			X	X
2.2.3 Evaluation				X
2.3 Dissemination Work				X
3. Second Year Report				X
Third Year				
4. Real-time sentiment analysis (Obj3)	X	X		
4.1 Build the online updatable semantic sentiment model	X	X		
4.2 Streaming data analysis using the modified cultural JST model	X	X		
4.3 Evaluation		X		
4.4 Dissemination Activities	X	X		
5. Thesis Writing			X	X

Table 4.1: Progress Plan for the Second and Third Year

Author's Publications

1. Hassan Saif, Yulan He, and Harith Alani. Semantic Smoothing for Twitter Sentiment Analysis. In Proceeding of the 10th International Semantic Web Conference (ISWC) (2011).
2. Hassan Saif, Yulan He, and Harith Alani. Alleviating Data Sparsity for Twitter Sentiment Analysis. In Proceedings, 2nd Workshop on Making Sense of Microposts (#MSM2012): in conjunction with WWW 2012
3. Yulan He and Hassan Saif. Quantising Opinons for Political Tweets Analysis.In Proceeding of the The eighth international conference on Language Resources and Evaluation (LREC) - In Submission (2012).

Conclusion

The emergence of microblogging services combined with the vast spread of social networking websites has established new phenomenon with millions of people sharing their thoughts and publishing their opinions everyday. Sentiment analysis of microblogs has been widely studied recently. However, unique characteristics of microblogs such as the language variations and the short length of users' post pose several challenges for sentiment classification problem over such noisy data. One challenge is data sparsity, others are open-domain and data dynamics.

Along this report, a concise study of the previous work on the domain of sentiment analysis of microblogs was presented and major gaps and challenges of this work were dissuaded. Research question followed by research objectives were outlined.

As a pilot work that has been conducted in the first year of my PhD, I have proposed a feature-based approach to alleviate the data sparsity problem of tweets data. Two sets of features have been used, semantic features and sentiment-topic features. Experimental results show that both methods improve upon the baseline model. Moreover, compared to the existing approaches to Twitter sentiment analysis, my approach are much more simple and yet attain comparable performance.

My research plan for the next two years will focus on (1) conducting more experiments in order to enhance the proposed model regarding about the data sparsity problem. (2) Aiming to address the data dynamics problem by doing the analysis at the use level where user's background knowledge will be modelled and incorporated into the sentiment classification process in order to facilitate real-time sentiment analysis of large-scale streaming data.

Bibliography

- [1] AGARWAL, A., XIE, B., VOVSHA, I., RAMBOW, O., AND PASSONNEAU, R. Sentiment analysis of twitter data. In *Proceedings of the ACL 2011 Workshop on Languages in Social Media* (2011), pp. 30–38.
- [2] ASSADI, D. Do religions influence customer behavior? confronting religious rules and marketing concepts. *Databases* 22 (2003), 10.
- [3] BARBOSA, L., AND FENG, J. Robust sentiment detection on twitter from biased and noisy data. In *Proceedings of COLING* (2010), pp. 36–44.
- [4] BHUIYAN, S. Social media and its effectiveness in the political reform movement in egypt. *Middle East Media Educator* 1, 1 (2011), 14–20.
- [5] BIFET, A., AND FRANK, E. Sentiment knowledge discovery in twitter streaming data. In *Discovery Science* (2010), Springer, pp. 1–15.
- [6] BOIY, E., HENS, P., DESCHACHT, K., AND MOENS, M. Automatic sentiment analysis in on-line text. In *Proceedings of the 11th International Conference on Electronic Publishing* (2007), pp. 349–360.
- [7] CHAOVALIT, P., AND ZHOU, L. Movie review mining: A comparison between supervised and unsupervised classification approaches.
- [8] CONOVER, M., RATKIEWICZ, J., FRANCISCO, M., GONCALVES, B., FLAMMINI, A., AND MENCZER, F. Political polarization on twitter. In *Proc. 5th Intl. Conference on Weblogs and Social Media* (2011).
- [9] DIAKOPOULOS, N., AND SHAMMA, D. Characterizing debate performance via aggregated twitter sentiment. In *Proceedings of the 28th international conference on Human factors in computing systems* (2010), ACM, pp. 1195–1198.
- [10] DWYER, S., MESAK, H., AND HSU, M. An exploratory examination of the influence of national culture on cross-national product diffusion. *Journal of International Marketing* (2005), 1–27.
- [11] GO, A., BHAYANI, R., AND HUANG, L. Twitter sentiment classification using distant supervision. *CS224N Project Report, Stanford* (2009).
- [12] GUERRA, P., CERF, L., PORTO, T., VELOSO, A., MEIRA JR, W., AND ALMEIDA, V. Exploiting temporal locality to determine user bias in microblogging platforms. *Journal of Information and Data Management* 2, 3 (2011), 273.

- [13] HATZIVASSILOGLOU, V., AND WIEBE, J. Effects of adjective orientation and gradability on sentence subjectivity. In *Proceedings of the 18th conference on Computational linguistics-Volume 1* (2000), Association for Computational Linguistics, pp. 299–305.
- [14] HE, Y., AND SAIF, H. Quantising Opinions for Political Tweets Analysis. In *Proceeding of the The eighth international conference on Language Resources and Evaluation (LREC) - In Submission* (2012).
- [15] HU, M., AND LIU, B. Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining* (2004), ACM, pp. 168–177.
- [16] HUSSAIN, M., AND HOWARD, P. the role of digital media. *Journal of Democracy* 22, 3 (2011), 35–48.
- [17] KOULOUMPIS, E., WILSON, T., AND MOORE, J. Twitter sentiment analysis: The good the bad and the omg! In *Proceedings of the ICWSM* (2011).
- [18] LI, S., AND ZONG, C. Multi-domain sentiment classification. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies: Short Papers* (2008), Association for Computational Linguistics, pp. 257–260.
- [19] LIN, C., AND HE, Y. Joint sentiment/topic model for sentiment analysis. In *Proceeding of the 18th ACM conference on Information and knowledge management* (2009), ACM, pp. 375–384.
- [20] LIVNE, A., SIMMONS, M., ADAR, E., AND ADAMIC, L. The party is over here: Structure and content in the 2010 election. In *Fifth International AAAI Conference on Weblogs and Social Media* (2011).
- [21] MEI, Q., LING, X., WONDRA, M., SU, H., AND ZHAI, C. Topic sentiment mixture: modeling facets and opinions in weblogs. In *Proceedings of the 16th international conference on World Wide Web* (2007), ACM, pp. 171–180.
- [22] NARAYANAN, R., LIU, B., AND CHOUDHARY, A. Sentiment Analysis of Conditional Sentences. In *EMNLP* (2009), pp. 180–189.
- [23] PAK, A., AND PAROUBEK, P. Twitter as a corpus for sentiment analysis and opinion mining. *Proceedings of LREC 2010* (2010).
- [24] PANG, B., AND LEE, L. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics* (2004), Association for Computational Linguistics, p. 271.
- [25] PANG, B., AND LEE, L. Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval* 2, 1-2 (2008), 1–135.

- [26] PANG, B., LEE, L., AND VAITHYANATHAN, S. Thumbs up?: sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10* (2002), Association for Computational Linguistics, pp. 79–86.
- [27] PENNEBAKER, J., BOOTH, R., AND FRANCIS, M. Liwc2007: Linguistic inquiry and word count. *Computer software*. Austin, TX: LIWC. net (2007).
- [28] PETROVIC, S., OSBORNE, M., AND LAVRENKO, V. The edinburgh twitter corpus. In *Proceedings of the NAACL HLT Workshop on Computational Linguistics in a World of Social Media* (2010), pp. 25–26.
- [29] PNG, I., TAN, B., AND WEE, K. Dimensions of national culture and corporate adoption of it infrastructure. *IEEE Transactions on Engineering Management*, 48, 1 (2001), 36–45.
- [30] READ, J., AND CARROLL, J. Weakly supervised techniques for domain-independent sentiment classification. In *Proceeding of the 1st international CIKM workshop on Topic-sentiment analysis for mass opinion* (2009), pp. 45–52.
- [31] SAIF, H., HE, Y., AND ALANI, H. Semantic Smoothing for Twitter Sentiment Analysis. In *Proceeding of the 10th International Semantic Web Conference (ISWC)* (2011).
- [32] SAIF, H., HE, Y., AND ALANI, H. Alleviating Data Sparsity for Twitter Sentiment Analysis. In *Proceedings, 2nd Workshop on Making Sense of Microposts (#MSM2012): Big things come in small packages: in conjunction with WWW 2012* (2012).
- [33] SPERIOSU, M., SUDAN, N., UPADHYAY, S., AND BALDRIDGE, J. Twitter polarity classification with label propagation over lexical links and the follower graph. *Proceedings of the EMNLP First workshop on Unsupervised Learning in NLP* (2011), 53–63.
- [34] TABOADA, M., AND GRIEVE, J. Analyzing appraisal automatically. In *Proceedings of AAAI Spring Symposium on Exploring Attitude and Affect in Text (AAAI Technical Report SS-04-07)* (2004), pp. 158–161.
- [35] TAN, C., LEE, L., TANG, J., JIANG, L., ZHOU, M., AND LI, P. User-level sentiment analysis incorporating social networks. *Arxiv preprint arXiv:1109.6018* (2011).
- [36] TSUR, O., DAVIDOV, D., AND RAPPOPORT, A. Icwsn-a great catchy name: Semi-supervised recognition of sarcastic sentences in online product reviews. *Proceeding of ICWSM* (2010).
- [37] TSYTSARAU, M., AND PALPANAS, T. Survey on mining subjective data on the web. *Data Mining and Knowledge Discovery* (2011), 1–37.
- [38] TUMASJAN, A., SPRENGER, T., SANDNER, P., AND WELPE, I. Predicting elections with twitter: What 140 characters reveal about political sentiment. In *Proceedings of the Fourth International AAAI Conference on Weblogs and Social Media* (2010), pp. 178–185.
- [39] TURNEY, P. Thumbs up or thumbs down? semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL'02)* (2002).

-
- [40] VAN EVERDINGEN, Y., AND WAARTS, E. The effect of national culture on the adoption of innovations. *Marketing Letters* 14, 3 (2003), 217–232.
- [41] WARD, J., AND OSTROM, A. The internet as information minefield:: An analysis of the source and content of brand information yielded by net searches. *Journal of Business research* 56, 11 (2003), 907–914.
- [42] WILSON, T., WIEBE, J., AND HOFFMANN, P. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing* (2005), Association for Computational Linguistics, pp. 347–354.
- [43] YANG, H., SI, L., AND CALLAN, J. Knowledge transfer and opinion detection in the trec2006 blog track. In *Proceedings of TREC* (2006), vol. 120, Citeseer.
- [44] YOON, E., GUFFEY, H., AND KIJEWski, V. The effects of information and company reputation on intentions to buy a business service. *Journal of Business Research* 27, 3 (1993), 215–228.
- [45] YU, H., AND HATZIVASSILOGLou, V. Towards answering opinion questions: Separating facts from opinions and identifying the polarity of opinion sentences. In *Proceedings of the 2003 conference on Empirical methods in natural language processing-Volume 10* (2003), Association for Computational Linguistics, pp. 129–136.
- [46] ZHAO, J., LIU, K., AND WANG, G. Adding redundant features for CRFs-based sentence sentiment classification. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)* (2008), pp. 117–126.