Sentiment Analysis in Social Media Texts

Alexandra Balahur
European Commission Joint Research Centre
Vie E. Fermi 2749
21027 Ispra (VA), Italy
`alexandra.balahur@jrc.ec.europa.eu`

Abstract

This paper presents a method for sentiment analysis specifically designed to work with Twitter data (tweets), taking into account their structure, length and specific language. The approach employed makes it easily extendible to other languages and makes it able to process tweets in near real time. The main contributions of this work are: a) the pre-processing of tweets to normalize the language and generalize the vocabulary employed to express sentiment; b) the use minimal linguistic processing, which makes the approach easily portable to other languages; c) the inclusion of higher order n-grams to spot modifications in the polarity of the sentiment expressed; d) the use of simple heuristics to select features to be employed; e) the application of supervised learning using a simple Support Vector Machines linear classifier on a set of realistic data. We show that using the training models generated with the method described we can improve the sentiment classification performance, irrespective of the domain and distribution of the test sets.

1 Introduction

Sentiment analysis is the Natural Language Processing (NLP) task dealing with the detection and classification of sentiments in texts. Usually, the classes considered are "positive", "negative" and "neutral", although in some cases finer-grained categories are added (e.g. "very positive" and "very negative") or only the "positive" and "negative" classes are taken into account. Another related task - emotion detection - concerns the classification of text into several classes of emotion, usually the basic ones, as described by Paul Ekman (Ekman, 1992). Although different in some ways, some of the research in the field has considered these tasks together, under the umbrella of sentiment analysis.

This task has received a lot of interest from the research community in the past years. The work done regarded the manner in which sentiment can be classified from texts pertaining to different genres and distinct languages, in the context of various applications, using knowledge-based, semi-supervised and supervised methods (Pang and Lee, 2008). The result of the analyses performed have shown that the different types of text require specialized methods for sentiment analysis, as, for example, sentiments are not conveyed in the same manner in newspaper articles and in blogs, reviews, forums or other types of user-generated contents (Balahur et al., 2010).

In the light of these findings, dealing with sentiment analysis in Twitter requires an analysis of the characteristics of such texts and the design of adapted methods.

Additionally, the sentiment analysis method employed has to consider the requirements of the final application in which it will be used. There is an important difference between deploying a system working for languages such as English, for which numerous linguistic resources and analysis tools exist and a system deployed for languages with few such tools or one that is aimed at processing data from a large set of languages. Finally, a sentiment analysis system working with large sets of data (such as the one found in Twitter) must be able to process texts fast. Therefore, using highly complex methods may delay producing useful results.

In the light of these considerations, this paper

presents a method for sentiment analysis that takes into account the special structure and linguistic content of tweets. The texts are pre-processed in order to normalize the language employed and remove noisy elements. Special usage of language (e.g. repeated punctuation signs, repeated letters) are marked as special features, as they contribute to the expressivity of the text in terms of sentiment. Further on, sentiment-bearing words, as they are found in three highly-accurate sentiment lexicons - General Inquirer (GI) (Stone et al., 1966), Linguistic Inquiry and Word Count (LIWC) (Tausczik and Pennebaker, 2010) and MicroWNOp (Cerini et al., 2007) - are replaced with unique labels, correspoding to their polarity. In the same manner, modifiers (negations, intensifiers and diminishers) are also replaced with unique labels representing their semantic class. Finally, we employ supervised learning with Support Vector Machines Sequential Minimal Optimization (SVM SMO) (Platt, 1998) using a simple, linear kernel (to avoid overfitting of data) and the unigrams and bigrams from the training set as features. We obtain the best results by using unique labels for the affective words and the modifiers, unigrams and bigrams as features and posing the condition that each feature considered in the supervised learning process be present in the training corpora at least twice.

The remainder of this article is structured as follows: Section 2 gives an overview of the related work. In Section 3, we present the motivations and describe the contributions of this work. In the following section, we describe in detail the process followed to pre-process the tweets and build the classification models. In Section 5, we present the results obtained using different datasets and combinations of features and discuss their causes and implications. Finally, Section 6 summarizes the main findings of this work and sketches the lines for future work.

2   Related Work

One of the first studies on the classification of polarity in tweets was (Go et al., 2009). The authors conducted a supervised classification study on tweets in English, using the emoticons (e.g. ":)", ":(", etc.) as markers of positive and negative tweets. (Read, 2005) employed this method to generate a corpus of positive tweets, with positive emoticons ":)", and negative tweets with negative emoticons ":(". Subsequently, they employ different supervised approaches (SVM, Naïve Bayes and Maximum Entropy) and various sets of features and conclude that the simple use of unigrams leads to good results, but it can be slightly improved by the combination of unigrams and bigrams.

In the same line of thinking, (Pak and Paroubek, 2010) also generated a corpus of tweets for sentiment analysis, by selecting positive and negative tweets based on the presence of specific emoticons. Subsequently, they compare different supervised approaches with n-gram features and obtain the best results using Naïve Bayes with unigrams and part-of-speech tags.

Another approach on sentiment analysis in tweet is that of (Zhang et al., 2011). Here, the authors employ a hybrid approach, combining supervised learning with the knowledge on sentiment-bearing words, which they extract from the DAL sentiment dictionary (Whissell, 1989). Their pre-processing stage includes the removal of retweets, translation of abbreviations into original terms and deleting of links, a tokenization process, and part-of-speech tagging. They employ various supervised learning algorithms to classify tweets into positive and negative, using n-gram features with SVM and syntactic features with Partial Tree Kernels, combined with the knowledge on the polarity of the words appearing in the tweets. The authors conclude that the most important features are those corresponding to sentiment-bearing words. Finally, (Jiang et al., 2011) classify sentiment expressed on previously-given "targets" in tweets. They add information on the context of the tweet to its text (e.g. the event that it is related to). Subsequently, they employ SVM and General Inquirer and perform a three-way classification (positive, negative, neutral).

3   Motivation and Contribution

As we have seen in the previous section, several important steps have already been taken into analyzing the manner in which sentiment can be automatically detected and classified from Twitter data. The research we described in previous section has already dealt with some of the issues that are posed by short,

informal texts, such as the tweets. However, these small snippets of text have several liguistic peculiarities that can be employed to improve the sentiment classification performance. We describe these peculiarities below:

- Tweets are short, user-generated text that may contain no more than 140 characters (strongly related to the standard 160-character length of SMS [1]). Users are marked with the "@" sign and topics with the "#" (hashtag) sign.

- In general, the need to include a large quantity of information in small limit of characters leads to the fact that tweets sometimes have no grammatical structure, contain misspellings and abbreviations.

- Some of the tweets are simply posted from the websites of news providers (news agencies, newspapers) and therefore they contain only titles of news. However, subjective tweets, in which users comment on an event, are highly marked by sentiment-bearing expressions, either in the form of affective words, or by employins specific modalities - e.g. the use of capital letters or repeated punctuation signs to stress upon specific words. Most of the times, these words are sentiment-bearing ones.

- The language employed in subjective tweets includes a specific slang (also called "urban expressions" [2]) and emoticons (graphical expressions of emotions through the use of punctuation signs).

- Most of the times, the topic that is discusses in the tweets is clearly marked using hashtags. Thus, there is no need to employ very complex linguistic tools to determine it.

- In major events, the rate of tweets per minute commenting or retweeting information surpasses the rate of thousands per minute.

- Twitter is available in more than 30 languages. However, users tweet in more than 80 languages. The information it contains can be useful to obtain information and updates about, for example, crisis events [3], in real time. In order to benefit from this, however, a system processing these texts has to be easily adaptable to other languages and it has to work in near real time.

Bearing this in mind, the main contributions we bring in this paper are:

1. The pre-processing of tweets to normalize the language and generalize the vocabulary employed to express sentiment. At this stage, we take into account the linguistic peculiarities of tweets, regarding spelling, use of slang, punctuation, etc., and also replace the sentiment-bearing words from the training data with a unique label. In this way, the sentence "I love roses." will be equivalent to the sentence "I like roses.", because "like" and "love" are both positive words according to the GI dictionary. If example 1 is contained in the training data and example 2 is contained in the test data, replacing the sentiment-bearing word with a general label increases the chance to have example 2 classified correctly. In the same line of thought, we also replaced modifiers with unique corresponding labels.

2. The use of minimal linguistic processing, which makes the approach easily portable to other languages. We employ only tokenization and do not process texts any further. The reason behind this choice is that we would like the final system to work in a similar fashion for as many languages as possible and for some of them, little or no tools are available.

3. The inclusion of bigrams to spot modifications in the polarity of the sentiment expressed. As such, we can learn general patterns of sentiment expression (e.g. "negation positive", "intensifier negative", etc.).

4. The use of simple heuristics to select features to be employed. Although feature selection algorithms are easy to apply when employing a data mining environment, the final choice is influenced by the data at hand and it is difficult to

---

[1] http://en.wikipedia.org/wiki/Twitter
[2] http://www.urbandictionary.com/

[3] http://blog.twitter.com/2012/10/hurricane-sandy-resources-on-twitter.html

employ on new sets of data. After performing various tests, we chose to select the features to be employed in the classification model based on the condition that they should occur at least once in the training set.

5. The application of supervised learning using a simple Support Vector Machines linear classifier on a set of realistic data.

We show that using the training models generated with the method described we can improve the sentiment classification performance, irrespective of the domain and distribution of the test sets.

4 Sentiment Analysis in Tweets

Our sentiment analysis system is based on a hybrid approach, which employs supervised learning with a Support Vector Machines Sequential Minimal Optimization (Platt, 1998) linear kernel, on unigram and bigram features, but exploiting as features sentiment dictionaries, emoticon lists, slang lists and other social media-specific features. We do not employ any specific language analysis software. The aim is to be able to apply, in a straightforward manner, the same approach to as many languages as possible. The approach can be extended to other languages by using similar dictionaries that have been created in our team. They were built using the same dictionaries we employ in this work and their corrected translation to Spanish. The new sentiment dictionaries were created by simultaneously translating from these two languages to a third one and considering the intersection of the trainslations as correct terms. Currently, new such dictionaries have been created for 15 other languages.

The sentiment analysis process contains two stages: pre-processing and sentiment classification.

4.1 Tweet Pre-processing

The language employed in Social Media sites is different from the one found in mainstream media and the form of the words employed is sometimes not the one we may find in a dictionary. Further on, users of Social Media platforms employ a special "slang" (i.e. informal language, with special expressions, such as "lol", "omg"), emoticons, and often emphasize words by repeating some of their letters.

Additionally, the language employed in Twitter has specific characteristics, such as the markup of tweets that were reposted by other users with "RT", the markup of topics using the "#" (hash sign) and of the users using the "@" sign.

All these aspects must be considered at the time of processing tweets. As such, before applying supervised learning to classify the sentiment of the tweets, we preprocess them, to normalize the language they contain. The pre-processing stage contains the following steps:

- Repeated punctuation sign normalization

  In the first step of the pre-processing, we detect repetitions of punctuation signs (".", "!" and "?"). Multiple consecutive punctuation signs are replaced with the labels "multistop", for the fullstops, "multiexclamation" in the case of exclamation sign and "multiquestion" for the question mark and spaces before and after.

- Emoticon replacement

  In the second step of the pre-processing, we employ the annotated list of emoticons from SentiStrength[4] and match the content of the tweets against this list. The emoticons found are replaced with their polarity ("positive" or "negative") and the "neutral" ones are deleted.

- Lower casing and tokenization.

  Subsequently, the tweets are lower cased and split into tokens, based on spaces and punctuation signs.

- Slang replacement

  The next step involves the normalization of the language employed. In order to be able to include the semantics of the expressions frequently used in Social Media, we employed the list of slang from a specialized site [5].

- Word normalization

  At this stage, the tokens are compared to entries in Rogets Thesaurus. If no match is found, repeated letters are sequentially reduced to two or one until a match is found in the dictionary (e.g.

"perrrrrrrrrrrrrrrrrrrfeeect" becomes "perrfeect", "perfeect", "perrfect" and subsequently "perfect"). The words used in this form are maked as "stressed".

- Affect word matching

  Further on, the tokens in the tweet are matched against three different sentiment lexicons: GI, LIWC and MicroWNOp, which were previously split into four different categories ("positive", "high positive", "negative" and "high negative"). Matched words are replaced with their sentiment label - i.e. "positive", "negative", "hpositive" and "hnegative". A version of the data without these replacements is also maintained, for comparison purposes.

- Modifier word matching

  Similar to the previous step, we employ a list of expressions that negate, intensify or diminish the intensity of the sentiment expressed to detect such words in the tweets. If such a word is matched, it is replaced with "negator", "intensifier" or "diminisher", respectively. As in the case of affective words, a version of the data without these replacements is also maintained, for comparison purposes.

- User and topic labeling

  Finally, the users mentioned in the tweet, which are marked with "@", are replaced with "PERSON" and the topics which the tweet refers to (marked with "#") are replaced with "TOPIC".

4.2 Sentiment Classification of Tweets

Once the tweets are pre-processed, they are passed on to the sentiment classification module. We employed supervised learning using SVM SMO with a linear kernel, based on boolean features - the presence or absence of n-grams (unigrams, bigrams and unigrams plus bigrams) determined from the training data (tweets that were previousely pre-processed as described above). Bigrams are used specifically to spot the influence of modifiers (negations, intensifiers, diminishers) on the polarity of the sentiment-bearing words. We tested the approach on different datasets and dataset splits, using the Weka data

mining software [6]. The training models are built on a cluster of computers (4 cores, 5000MB of memory each). However, the need for such extensive resources is only present at the training stage. Once the feature set is determined and the models are built using Weka, new examples must only be represented based on the features extracted from the training set and the classification is a matter of miliseconds.

The different evaluations scenarios and results are presented in the following section.

5 Evaluation and Discussion

Although the different steps included to eliminate the noise in the data and the choice of features have been refined using our in-house gathered Twitter data, in order to evaluate our approach and make it comparable to other methods, we employ three different data sets, which are described in detail in the following subsections.

5.1 Data Sets

- SemEval 2013 Data

  The first one is the data provided for training for the upcoming SemEval 2013 Task 2 "Sentiment Analysis from Twitter" [7]. The initial training data has been provided in two stages: 1) sample datasets for the first task and the second task and 2) additional training data for the two tasks. We employ the joint sample datasets as test data (denoted as $t*$) and the data released subsequently as training data (denoted as $T*$). We employ the union of these two datasets to perform cross-validation experiments (the joint dataset is denoted as $T * +t*$. The characteristics of the dataset are described in Table 1. On the last column, we also include the baseline in terms of accuracy, which is computed as the number of examples of the majoritary class over the total number of examples:

- Set of tweets labeled with basic emotions.

  The set of emotion-annotated tweets by (Mohammad, 2012), which we will denote as $TweetEm$. It contains 21051 tweets annotated according to the Ekman categories of ba-

---

sic emotion - anger, disgust, fear, joy, sadness, surprise. We employ this dataset to test the results of our best-performing configurations on the test set. This set contains a total of 21051 tweets (anger - 1555, disgust - 761, fear - 2816, joy - 8240, sadness - 3830, surprise - 3849). As mentioned in the paper by (Mohammad, 2012), a system that would guess the classes, would perfom at aroung 49.9% accuracy.

- Set of short blog sentences labeled with basic emotions.

  The set of blog sentences employed by (Aman and Szpakowicz, 2007), which are annotated according to the same basic emotions identified by Paul Ekman, with the difference that the "joy" category is labeled as "happy". This test set contains also examples which contain no emotions. These sentences were removed. We will denote this dataset as $BlogEm$. This set contains 1290 sentences annotated with emotion (anger - 179, disgust - 172, fear - 115, joy - 536, sadness - 173, surprise - 115). We can consider as baseline the case in which all the examples are assigned to the majority class (joy), which would lead to an accuracy of 41.5%.

Data	#Tweet	#Pos.	#Neg.	#Neu.	Bl%
T*	19241	4779	2343	12119	62
t*	2597	700	393	1504	57
T*+t*	21838	5479	2736	13623	62

Table 1: Characteristics of the training (T*), testing (t*) and joint training and testing datasets.

5.2 Evaluation and Results

In order to test our sentiment analysis approach, we employed the datasets described above. In the case of the SemEval data, we performed an exhaustive evaluation of the possible combination of features to be employed. We tested the entire dataset of tweets (T*+t*) using 10-fold cross-validation. The first set of evaluations concerned the use of the pre-processed tweets in which the affective words and modifiers were have not been replaced. The combination of features tested were: unigrams (U), bigrams (B), unigrams and bigrams together ($U + B$)

and unigrams and bigrams together, selecting only the features that appear at least twice in the data ($U + B + FS$). The second set of evaluations aimed at quantifying the difference in performance when the affective words and the modifiers were replaced with generic labels. We tested the best performing approaches from the first set of evaluations ($U + B$ and $U + B + FS$), by replacing the words that were found in the affect dictionaries and the modifiers with their generic labels. These evaluations are denoted as $U + B + D$ and $U + B + D + FS$. The results of these evaluations are shown in Table 2.

Features	10-f-CV T*+t*
U	71.82
B	66.30
$U + B$	82.01
$U + B + D$	81.15
$U + B + FS$	74.00
$U + B + D + FS$	85.07

Table 2: Results in terms of accuracy for 10-fold cross-validation using different combinations of features for the sentiment classification of tweets on the entire set of SemEval 2013 training data.

The same experiments are repeated by employing T* as training data and t* as test data. The aim of these experiments is to test how well the method can perform on new data. The results of these evaluations are shown in Table 3. In order to test if in-

Features	Train(T*) & test(t*)
U	74.90
B	63.27
$U + B$	77.00
$U + B + D$	76.45
$U + B + FS$	75.69
$U + B + D + FS$	79.97

Table 3: Results in terms of accuracy for the different combination of features for the sentiment classification of tweets, using T* as training and t* as test set.

deed the use of sentiment dictionaries, modifiers and the simple feature selection method improves on the best performing approach that does not employ these additional features, we tested both the approaches on the $TweetEm$ and $BlogEm$ datasets. In this case,

however, the classification is done among 6 different classes of emotions. Although the results are lower(as it can be seen in Table 4, they are comparable to those obtained by (Mohammad, 2012) (when using $U+B$) and show an improvement when using the affect dictionaries and simple feature selection. They also confirm the fact that the best performance on the data is obtained replacing the modifiers and the words found in affect dictionaries with generic labels, using unigrams and bigrams as and eliminating those n-grams that appear only once.

Features	Tweet Em	Blog Em
$U+B$	49.00	51.08
$U+B+D+FS$	51.08	53.70

Table 4: Results in terms of accuracy for the different combination of features for the emotion classification of tweets and short blog sentences.

The results obtained confirm that the use of unigram and bigram features (appearing at least twice) with generalized affective words and modifiers obtains the best results. Although there is a significant improvement in the accuracy of the classification, the most important difference in the classification performance is given by the fact that using this combination, the classifier is no longer biased by the class with the highest number of examples. We can notice this for the case of tweets, for which the confusion matrices are presented in Table 5 and Table 6. In the table header, the correspondence is: a = joy, b = fear, c = surprise, d = anger, e = disgust, f = sadness. In the first case, the use of unigrams and bigrams leads to the erroneous classification of examples to the majority class. When employing the features in which affective words and modifiers have been replaced with generic labels, the results are not only improved, but they classifier is less biased towards the majority class. In this case, the incorrect assignments are made to classes that are more similar in vocabulary (e.g. anger - disgust, anger - sadness). In the case of surprise, examples relate both to positive, as well as negative surprises. Therefore, there is a similarity in the vocabulary employed to both these classes.

	a	b	c	d	e	f
a	5879	178	865	246	349	723
b	657	1327	339	67	59	367
c	1243	248	1744	123	129	362
d	549	189	79	419	48	271
e	167	55	45	89	160	245
f	570	405	611	625	233	1386

Table 5: Confusion matrix for the emotion classification of the $TweetEm$ dataset employing the sentiment dictionaries.

	a	b	c	d	e	f
a	6895	252	395	57	20	622
b	1384	861	207	49	11	302
c	1970	147	1258	39	13	421
d	884	133	88	101	18	332
e	433	54	60	32	40	142
f	2097	192	287	72	23	1160

Table 6: Confusion matrix for the emotion classification of the $TweetEm$ dataset without employing the sentiment dictionaries.

5.3 Discussion

From the results obtained, we can conclude that, on the one hand, the best features to be employed in sentiment analysis in tweets are unigrams and bigrams together. Secondly, we can see that the use of generalizations, by employing unique labels to denote sentiment-bearing words and modifiers highly improves the performance of the sentiment classification. The usefulness of pre-processing steps is visible from the fact that among the bigrams that were extracted from the training data we can find the unique labels employed to mark the use of repeated punctuation signs, stressed words, affective words and modifiers and combinations among them. Interesting bigrams that were discovered using these generalizations are, e.g. "negative multiexclamation", "positive multiexclamation", "positive multistop" - which is more often found in negative tweets -,"negator positive", "diminisher positive", "mostly diminisher", "hnegative feeling", "hnegative day", "eat negative","intensifier hnegative". All these extracted features are very useful to detect and classify sentiment in tweets and most of them would be ignored if the vocabulary were different in the train-

ing and test data or if, for example, a stressed word would be written under different forms or a punctuation sign would be repeated a different number of times. We can see that the method employed obtains good results, above the ones reported so far with the state-of-the-art approaches. We have seen that the use of affect and modifier lexica generalization has an impact on both the quantitative performance of the classification, as well as on the quality of the results, making the classifier less biased towards the class with a significantly larger number of examples. In practice, datasets are not balanced, so it is imporant that a classifier is able to assign (even incorrectly) an example to a class that is semantically similar and not to a class with totally opposite affective orientation. In this sense, as we have seen in the detailed results obtained on the $TweetEm$ dataset, it is preferable that, e.g. the examples pertaining to the emotion classes of anger and sadness are mistakenly classified as the other. However, it is not acceptable to have such a high number of examples from these classes labeled as "joy". Finally, by inspecting some of the examples in the three datasets, we noticed that a constant reason for error remains the limited power of the method to correctly spot the scope of the negations and modifiers. As such, we plan to study the manner in which skip-bigrams (bigrams made up of non-consecutive tokens) can be added and whether or not they will contribute to (at least partially) solve this issue.

6   Conclusions and Future Work

In this article, we presented a method to classify the sentiment in tweets, by taking into account their peculiarities and adapting the features employed to their structure and content. Specifically, we employed a pre-processing stage to normalize the language and generalize the vocabulary employed to express sentiment. This regarded spelling, slang, punctuation, etc., and the use of sentiment dictionaries and modifier lists to generalize the patterns of sentiment expression extracted from the training data. We have shown that the use of such generalized features significantly improves the results of the sentiment classification,when compared to the best-performing approaches that do not use affect dictionaries. Additionally, we have shown that we can

obtain good results even though we employ minimal linguistic processing. The advantage of this approach is that it makes the method easily applicable to other languages. Finally, we have shown that the use of a simple heuristic, concerning filtering out features that appear only once, improves the results. As such, the method is less dependent on the dataset on which the classification model is trained and the vocabulary it contains. Finally, we employed a simple SVM SMO linear classifier to test our approach on three different data sets. Using such an approach avoids overfitting the data and, as we have shown, leads to comparable performances on different datasets. In future work, we plan to evaluate the use of higher-order n-grams (3-grams) and skip-grams to extract more complex patterns of sentiment expressions and be able to identify more precisely the scope of the negation. Additionally, we plan to evaluate the influence of deeper linguistic processing on the results, by performing stemming, lemmatizing and POS-tagging. Further on, we would like to extend our approach on generalizing the semantic classes of words and employing unique labels to group them (e.g. label mouse, cat and dog as "animal"). Finally, we would like to study the performance of our approach in the context of tweets related to specific news, in which case these short texts can be contextualized by adding further content from other information sources.

References

Saima Aman and Stan Szpakowicz. 2007. Identifying expressions of emotion in text. In *Proceedings of the 10th international conference on Text, speech and dialogue*, TSD'07, pages 196–205, Berlin, Heidelberg. Springer-Verlag.

Alexandra Balahur, Ralf Steinberger, Mijail Kabadjov, Vanni Zavarella, Erik van der Goot, Matina Halkia, Bruno Pouliquen, and Jenya Belyaeva. 2010. Sentiment analysis in the news. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Bente Maegaard, Joseph Mariani, Jan Odijk, Stelios Piperidis, Mike Rosner, and Daniel Tapias, editors, *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta, may. European Language Resources Association (ELRA).

S. Cerini, V. Compagnoni, A. Demontis, M. Formentelli, and G. Gandini, 2007. *Language resources and lin-*

guistic theory: Typology, second language acquisition, English linguistics., chapter Micro-WNOp: A gold standard for the evaluation of automatically compiled lexical resources for opinion mining. Franco Angeli Editore, Milano, IT.

Paul Ekman. 1992. An argument for basic emotions. *Cognition & Emotion*, 6(3-4):169–200, May.

Alec Go, Richa Bhayani, and Lei Huang. 2009. Twitter sentiment classification using distant supervision. *Processing*, pages 1–6.

Long Jiang, Mo Yu, Ming Zhou, Xiaohua Liu, and Tiejun Zhao. 2011. Target-dependent twitter sentiment classification. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, pages 151–160, Stroudsburg, PA, USA. Association for Computational Linguistics.

Saif Mohammad. 2012. #emotional tweets. In **SEM 2012: The First Joint Conference on Lexical and Computational Semantics – Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*, pages 246–255, Montréal, Canada, 7-8 June. Association for Computational Linguistics.

Alexander Pak and Patrick Paroubek. 2010. Twitter as a corpus for sentiment analysis and opinion mining. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Bente Maegaard, Joseph Mariani, Jan Odijk, Stelios Piperidis, Mike Rosner, and Daniel Tapias, editors, *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10)*, Valletta, Malta; ELRA, may. European Language Resources Association. 19-21.

Bo Pang and Lillian Lee. 2008. Opinion mining and sentiment analysis. *Found. Trends Inf. Retr.*, 2(1-2):1–135, January.

John C. Platt. 1998. Sequential minimal optimization: A fast algorithm for training support vector machines. Technical report, Advances in Kernel Methods - Support Vector Learning.

Jonathon Read. 2005. Using emoticons to reduce dependency in machine learning techniques for sentiment classification. In *Proceedings of the ACL Student Research Workshop*, ACLstudent '05, pages 43–48, Stroudsburg, PA, USA. Association for Computational Linguistics.

Philip J. Stone, Dexter C. Dunphy, Marshall S. Smith, and Daniel M. Ogilvie. 1966. *The General Inquirer: A Computer Approach to Content Analysis*. MIT Press.

Yla R. Tausczik and James W. Pennebaker. 2010. The Psychological Meaning of Words: LIWC and Computerized Text Analysis Methods. *Journal of Language and Social Psychology*, 29(1):24–54, March.

Cynthia Whissell. 1989. The Dictionary of Affect in Language. In Robert Plutchik and Henry Kellerman, editors, *Emotion: theory, research and experience*, volume 4, The measurement of emotions. Academic Press, London.

Ley Zhang, Riddhiman Ghosh, Mohamed Dekhil, Meichun Hsu, and Bing Liu. 2011. Combining lexicon-based and learning-based methods for twitter sentiment analysis. Technical Report HPL-2011-89, HP, 21/06/2011.