Journal of Artificial Intelligence Research 1 (2014) 1-15 Submitted 12/13; published ?

State-of-the-Art in Sentiment Analysis of Short Informal Texts

Svetlana Kiritchenko Svetlana.Kiritchenko@nrc-cnrc.gc.ca
Xiaodan Zhu Xiaodan.Zhu@nrc-cnrc.gc.ca
Saif M. Mohammad Saif.Mohammad@nrc-cnrc.gc.ca
National Research Council Canada
1200 Montreal Rd., Ottawa, ON, Canada

Abstract

We describe a state-of-the-art sentiment analysis system that detects the sentiment of short informal textual messages such as tweets and SMS (message-level task). The system can also detect the sentiment of a word or a phrase within a message (term-level task). We adopt a supervised statistical text classification approach leveraging a variety of surface-form, semantic, and sentiment features. Our system was ranked first in a recent international competition SemEval-2013 on Sentiment Analysis in Twitter in both tasks, obtaining an F-score of 69.02 in the message-level task and 88.93 in the term-level task. Post-competition improvements to the system push the state-of-the-art to an F-score of 70.45 (message-level task) and 89.50 (term-level task). The system also obtains state-of-the-art performance on two additional datasets: the SemEval-2013 SMS test set and a corpus of movie review excerpts.

We perform extensive experiments to identify the contributions of various feature groups. The primary competitive advantage of our system comes from the use of two high-coverage tweet-specific sentiment lexicons. We automatically generated these lexicons from tweets with sentiment-word hashtags and from tweets with emoticons. Another advantage is in the way in which we handle negations. A common approach to negation handling is to reverse polarities of the words occurring in a negated context, even though that may not always capture the overall sentiment faithfully. We propose an empirical method to estimate the sentiment of words in negated contexts by creating a separate sentiment lexicon for negated words. The use of the lexicons for negated contexts and the lexicons for affirmative contexts results in performance gains of up to 6.5 percentage points over the gains obtained through the use of manually created general-purpose lexicons and other features. The automatic lexicons are made freely available.

1. Introduction

Sentiment Analysis involves determining the evaluative nature of a piece of text. For example, a product review can express a positive, negative, or neutral sentiment (or polarity). Automatically identifying sentiment expressed in text has a number of applications, including tracking sentiment towards products, movies, politicians, etc. (Pang & Lee, 2008; Mohammad & Yang, 2011), improving customer relation models (Bougie, Pieters, & Zeelenberg, 2003), detecting happiness and well-being (Schwartz, Eichstaedt, Kern, Dziurzynski, Lucas, Agrawal, Park, Lakshmikanth, Jha, Seligman, & Ungar, 2013), and improving automatic dialogue systems (Velásquez, 1997; Ravaja, Saari, Turpeinen, Laarni, Salminen, & Kivikangas, 2006). In the past decade, sentiment analysis techniques have been applied to

various types of text, e.g., user product reviews, newspaper headlines, email, and blogs. More recently, there has been substantial growth in the use of microblogging websites such as Twitter and in the use of Short Message Service (SMS) messages, especially in the developing world. Thus, there is now tremendous interest in sentiment analysis of tweets and SMS across a variety of domains such as commerce (Jansen, Zhang, Sobel, & Chowdury, 2009), health (Chew & Eysenbach, 2010; Salathé & Khandelwal, 2011), and disaster management (Verma, Vieweg, Corvey, Palen, Martin, Palmer, Schram, & Anderson, 2011; Mandel, Culotta, Boulahanis, Stark, Lewis, & Rodrigue, 2012).

Short informal textual messages such as tweets and SMS messages bring in new challenges to sentiment analysis. They are limited in length, usually spanning one sentence or less. They tend to have many misspellings, slang terms, and shortened forms of words. They also have special markers such as hashtags that are used to facilitate search, but can also indicate a topic or sentiment.

To promote research in sentiment analysis of short informal texts and to establish a common ground for comparison of different approaches, an international competition was organized by the Conference on Semantic Evaluation Exercises (SemEval-2013) (Wilson, Kozareva, Nakov, Rosenthal, Stoyanov, & Ritter, 2013).[1] This competition, officially referred to as *Sentiment Analysis in Twitter*, had two tasks. In the *message-level task*, the systems had to identify if a textual message as a whole expressed positive, negative, or neutral sentiment. In the *term-level task*, a target term (a single word or a multi-word expression) was marked in a message, and the goal was to identify the sentiment (positive, negative, or neutral) of that target term in the context of the message. For example, the word *unpredictable* expresses positive sentiment in sentence "*The movie has an unpredictable ending*"; whereas, it expresses negative sentiment in sentence "*The car has an unpredictable steering*". The organizers created and shared tweets for training, development, and testing. They also provided a second test set consisting of SMS messages. The purpose of having this out-of-domain test set was to assess the ability of the systems trained on tweets to generalize to other types of short informal texts. The competition attracted 44 teams; there were 48 submissions from 34 teams in the message-level task and 29 submissions from 23 teams in the term-level task. Our team, NRC-Canada, placed first in both tasks on the tweet test set, obtaining a macro-averaged F-score of 69.02 in the message-level task and 88.93 in the term-level task. Post-competition improvements to the system push the state-of-the-art to an F-score of 70.45 (message-level task) and 89.50 (term-level task). We also applied our classifier on the SMS test set without any further tuning. Nonetheless, the classifier still obtained the first position in identifying sentiment of SMS messages (F-score of 68.46) and the second position in detecting the sentiment of terms within SMS messages (F-score of 88.00; only 0.39 points behind the first-ranked system). With post-competition improvements (but still with no tuning specific to SMS), the system achieves an F-score of 69.77 in the message-level task and an F-score of 88.20 in the term-level task on that test set.

In addition, we evaluated the performance of our sentiment analysis system on the domain of movie review excerpts (message-level task only). The system was re-trained on the collection of about 7,800 positive and negative sentences extracted from movie reviews.

1. SemEval is an international forum for natural-language shared tasks. The competition we refer to is SemEval-2013 Task 2 (http://www.cs.york.ac.uk/semeval-2013/task2).

When applied on the test set of unseen sentences, the system was able to correctly classify 85.5% of the test set. This result exceeds the best result obtained on this dataset by a recursive deep learning approach that requires access to sentiment labels of all syntactic phrases in the training-data sentences (Socher, Perelygin, Wu, Chuang, Manning, Ng, & Potts, 2013). For the message-level task, we do not make use of sentiment labels of phrases in the training data, as that is often unavailable in real-world applications.

In this paper, we describe our state-of-the-art sentiment analysis system applied to both message-level and term-level tasks.[2] We adopt a supervised statistical text classification approach leveraging a variety of surface-form, semantic, and sentiment features. Given only limited amounts of training data, statistical sentiment analysis systems often benefit from the use of manually or automatically built sentiment lexicons. *Sentiment lexicons* are lists of words (and phrases) with prior associations to positive and negative sentiments. Some lexicons can additionally provide a sentiment score for a term to indicate its strength of evaluative intensity. Higher scores indicate greater intensity. For example, an entry *great (positive, 1.2)* states that the word *great* has positive polarity with the sentiment score of 1.2. An entry *acceptable (positive, 0.1)* specifies that the word *acceptable* has positive polarity and its intensity is lower than that of the word *great*.

In our sentiment analysis system, we utilized three freely available, manually created, general-purpose sentiment lexicons. In addition, we generated two high-coverage tweet-specific sentiment lexicons from about 2.5 million tweets using sentiment markers within them. These lexicons automatically capture many peculiarities of the social media language such as common intentional and unintentional misspellings (e.g., *gr8*, *lovin*, *coul*, *holys**t*), elongations (e.g., *yesssss*, *mmmmmmm*, *uugghh*), and abbreviations (e.g., *lmao*, *wtf*). They also include words that are not usually considered expressing sentiment, but that are often associated with positive/negative feelings (e.g., *party*, *birthday*, *homework*).

Sentiment lexicons provide knowledge on *prior* polarity (positive, negative, or neutral) of a word, i.e., its polarity in most contexts. However, in a particular context this prior polarity can change. One such obvious contextual sentiment modifier is negation. In a negated context, many words change their polarity or at least the evaluative intensity. For example, the word *good* is often used to express positive attitude whereas the phrase *not good* is clearly negative; the word *terrible* conveys a strong negative sentiment whereas the phrase *wasn't terrible* is only mildly negative. A conventional way of addressing negation in sentiment analysis is to reverse the polarity of a word, i.e. change a word's sentiment score s to $-s$ (Kennedy & Inkpen, 2005; Choi & Cardie, 2008). However, several studies have pointed out the inadequacy of this solution (Kennedy & Inkpen, 2006; Taboada, Brooke, Tofiloski, Voll, & Stede, 2011). We will show through experiments in Section 4.3 that many positive terms, though not all, tend to reverse their polarity when negated, whereas most negative terms remain negative and only change their evaluative intensity. Also, the degree of the intensity shift varies from term to term for both positive and negative terms. To accurately capture the effects of negation on different terms, we propose a corpus-based statistical approach to estimate sentiment scores of individual terms in the presence of negation. We

2. Our official SemEval submissions have been described in (Mohammad, Kiritchenko, & Zhu, 2013). This paper presents an extended version of the system. It includes the description, analysis, and evaluation of the novel lexicons for affirmative contexts and the lexicons for negated contexts. Also, it describes the experiments on applying the system to the domain of movie review excerpts.

build two lexicons: one for words in negated contexts (*Negated Context Lexicon*) and one for words in affirmative (non-negated) contexts (*Affirmative Context Lexicon*). Each word (or phrase) now has two scores, one in the Negated Context Lexicon and one in the Affirmative Context Lexicon. When analyzing the sentiment of a textual message, we use scores from the Negated Context Lexicon for words appearing in a negated context and scores from the Affirmative Context Lexicon for words appearing in an affirmative context.

We describe how we created the automatic, tweet-specific lexicons and demonstrate their superior predictive power over several manually and automatically created general-purpose lexicons in both supervised and unsupervised settings. Furthermore, our analysis reveals that these automatically built lexicons gave our system the competitive advantage in SemEval-2013. The use of the new lexicons results in gains of up to 6.5 percentage points over the gains obtained through the use of other features. Finally, we show that the lexicons built specifically for negated contexts better model negation than the reversing polarity approach.

The paper is organized as follows. We begin with a description of related work in Section 2. Next, we describe the sentiment analysis task and the data used in this research (Section 3). Section 4 presents the sentiment lexicons used in our system: existing manually created, general-purpose lexicons (Section 4.1) and our automatic, tweet-specific lexicons (Section 4.2). The lexicons built for affirmative and negated contexts are described in Section 4.3. The detailed description of our supervised sentiment analysis system, including the classification method and the feature sets, is presented in Section 5. Section 6 provides the results of the evaluation experiments. First, we evaluate the newly created lexicons in unsupervised settings (Section 6.1). The purpose of these experiments is to compare the predictive capacity of the individual lexicons without influence of other factors. Then, in Section 6.2 we assess the performance of the entire supervised system and examine the contribution of the features derived from our lexicons to the overall performance. Finally, we conclude and present directions for future work in Section 7.

All automatic lexicons described in this paper are made available to the research community.[3] Our sentiment analysis system can be replicated using freely available resources.

2. Related Work

Over the last decade, there has been an explosion of work exploring various aspects of sentiment analysis: detecting subjective and objective sentences; classifying sentences as positive, negative, or neutral; detecting the person expressing the sentiment and the target of the sentiment; detecting emotions such as joy, fear, and anger; visualizing sentiment in text; and applying sentiment analysis in health, commerce, and disaster management. Surveys by Pang and Lee (2008) and Liu and Zhang (2012) give a summary of many of these approaches.

Apart from sentiment lexicon features (mentioned in Section 1), some of the features commonly used by sentiment analysis systems include word and character ngrams, parts of speech, punctuations (!, ???), and word elongations (*hugggs, ahhhh*). Word and character ngrams are widely used in a number of text classification problems, and it is not surprising to find that they are beneficial for sentiment analysis as well. There are a few manually

3. www.purl.com/net/sentimentoftweets

created sentiment resources that have been successfully applied in sentiment analysis. The General Inquirer (GI) has sentiment labels for about 3,600 terms (Stone, Dunphy, Smith, Ogilvie, & associates, 1966). Hu and Liu (2004) manually labeled about 6,800 words and used them for detecting sentiment of customer reviews. The MPQA Subjectivity Lexicon, which draws from the General Inquirer and other sources, has sentiment labels for about 8,000 words (Wilson, Wiebe, & Hoffmann, 2005). The NRC Emotion Lexicon has sentiment and emotion labels for about 14,000 words (Mohammad & Turney, 2010). These labels were compiled through Mechanical Turk annotations.[4]

Semi-supervised and automatic methods have also been proposed to detect the polarity of words. Hatzivassiloglou and McKeown (1997) proposed an algorithm to determine the polarity of adjectives. SentiWordNet (SWN) was created using supervised classifiers as well as manual annotation (Esuli & Sebastiani, 2006). Turney and Littman (2003) proposed a minimally supervised algorithm to calculate the polarity of a word by determining if its tendency to co-occur with a small set of positive seed words is greater than its tendency to co-occur with a small set of negative seed words. Mohammad, Dunne, and Dorr (2009) automatically generated a sentiment lexicon of more than 60,000 words from a thesaurus. We used several of these lexicons in our system. In addition, we created two new sentiment lexicons from tweets using hashtags and emoticons. In Section 6, we show that these tweet-specific lexicons have a higher coverage and a stronger predictive power than the lexicons mentioned earlier.

Sentiment analysis systems have been applied to many different kinds of texts including customer reviews, news paper headlines (Bellegarda, 2010), novels (Boucouvalas, 2002; John, Boucouvalas, & Xu, 2006; Francisco & Gervás, 2006; Mohammad & Yang, 2011), emails (Liu, Lieberman, & Selker, 2003; Mohammad & Yang, 2011), blogs (Neviarouskaya, Prendinger, & Ishizuka, 2011; Genereux & Evans, 2006; Mihalcea & Liu, 2006), and tweets (Mohammad, 2012). Often these systems have to cater to the specific needs of the text such as formality versus informality, length of utterances, etc. Sentiment analysis systems developed specifically for tweets include those by Pak and Paroubek (2010), Agarwal, Xie, Vovsha, Rambow, and Passonneau (2011), Thelwall, Buckley, and Paltoglou (2011), Brody and Diakopoulos (2011), Aisopos, Papadakis, Tserpes, and Varvarigou (2012), Bakliwal, Arora, Madhappan, Kapre, Singh, and Varma (2012).

Some of the sentiment analysis systems benefit from the peculiarities of the text. For example, Go, Bhayani, and Huang (2009) use tweets with emoticons as labeled data for supervised training. Emoticons such as `:)` are considered positive labels of the tweet and emoticons such as `:(` are used as negative labels. Davidov, Tsur, and Rappoport (2010) and Kouloumpis, Wilson, and Moore (2011) use certain seed hashtag words such as *#cute* and *#sucks* as labels of positive and negative sentiment. Mohammad (2012) developed a classifier to detect emotions using tweets with emotion word hashtags (e.g., *#anger*, *#surprise*) as labeled data. In our system too, we make use of the emoticons and hashtag words as signals of positive and negative sentiment. We collected 775,000 sentiment-word hashtagged tweets and used 1.6 million emoticon tweets collected by Go et al. (2009). We generated sentiment lexicons from these datasets and used them (along with a relatively

4. https://www.mturk.com/mturk/welcome

small hand-labeled training dataset) to train a supervised classifier. We show that these sentiment lexicons are extremely helpful in sentiment analysis.

Negation plays an important role in determining sentiment. Automatic negation handling involves identifying a negation word such as *not*, determining the scope of negation (which words are affected by the negation word), and finally appropriately capturing the impact of the negation. (See work by Jia, Yu, and Meng (2009), Wiegand, Balahur, Roth, Klakow, and Montoyo (2010), Lapponi, Read, and Ovrelid (2012) for detailed analyses of negation handling.) Traditionally, the negation word is determined from a small hand-crafted list (Taboada et al., 2011). The scope of negation is often assumed to begin from the word following the negation word until the next punctuation mark or the end of the sentence (Polanyi & Zaenen, 2004; Kennedy & Inkpen, 2005). More sophisticated methods to detect the scope of negation through semantic parsing have also been proposed (Li, Zhou, Wang, & Zhu, 2010).

A common way to capture the impact of negation is to reverse the polarities of the sentiment words in the scope of negation (Kennedy & Inkpen, 2005; Choi & Cardie, 2008). However, as we already pointed out in the previous section, such a transformation is not always an appropriate representation. Trying to address this issue, Taboada et al. (2011) proposed to shift the sentiment score of a term in a negated context towards the opposite polarity by a fixed amount. Still, in their experiments the shift-score model did not agree with human judgment in many cases, especially for negated negative terms.

Recently, more complex approaches, such as recursive deep models, have been introduced (Socher, Huval, Manning, & Ng, 2012; Socher et al., 2013). The recursive deep models work in a bottom-top fashion over a parse-tree structure of a sentence to infer the sentiment label of the sentence as a composition of the sentiment expressed by its constituting parts: words and phrases. These models do not require any hand-crafted features or semantic knowledge, such as a list of negation words. However, they are computationally intensive and need substantial additional annotations (word and phrase-level sentiment labeling) to produce competitive results (Socher et al., 2013).

In this paper, we propose a simple corpus-based statistical method to determine the impact of negation on sentiment words. We create a new separate sentiment lexicon for negated words that we make freely available. We also analyze the impact of negation on sentiment scores of common sentiment terms.

Presently, we focus on sentiment analysis alone and do not attempt to associate the sentiment with its targets. There has been interesting work studying the latter problem (e.g., Jiang, Yu, Zhou, Liu, & Zhao, 2011; Sauper & Barzilay, 2013). We believe our approach can be incorporated into models that seek to identify the sentiment for a specified target.

3. Task and Data Description

In this work, we follow the definition of the task and use the data provided for the SemEval-2013 competition: Sentiment Analysis in Twitter (Wilson et al., 2013). This competition had two tasks: a message-level task and a term-level task. The objective of the *message-level task* is to detect whether the whole message conveys a positive, negative, or neutral sentiment. The objective of the *term-level task* is to detect whether a given target term (a single

Table 1: Class distributions in the SemEval-2013 training set, development set and two testing sets.

Dataset	Positive	Negative	Neutral	Total
Message-level task:				
Training set	3,045 (37%)	1,209 (15%)	4,004 (48%)	8,258
Development set	575 (35%)	340 (20%)	739 (45%)	1,654
Tweet test set	1,572 (41%)	601 (16%)	1,640 (43%)	3,813
SMS test set	492 (23%)	394 (19%)	1,208 (58%)	2,094
Term-level task:				
Training set	4,831 (62%)	2,540 (33%)	385 (5%)	7,756
Development set	648 (57%)	430 (38%)	57 (5%)	1,135
Tweet test set	2,734 (62%)	1,541 (35%)	160 (3%)	4,435
SMS test set	1,071 (46%)	1,104 (47%)	159 (7%)	2,334

word or a multi-word expression) conveys a positive, negative, or neutral sentiment in the context of a message. Note that the same term may express different sentiments in different contexts (e.g., *unpredictable ending* (positive) and *unpredictable steering* (negative)).

Two test sets - one with tweets and one with SMS messages - were provided to the participants for each task. Training and development data were available only for tweets. Here we briefly describe how the data were collected and annotated (for more details see the task description paper (Wilson et al., 2013)). The organizers identified certain frequently mentioned entities, such as Obama, Leonard Cohen, Superbowl, and gathered tweets with these entities through the public streaming Twitter API for a period of one year: from January 2012 to January 2013. To reduce the data skew towards the neutral class, messages that did not contain any polarity word listed in SentiWordNet 3.0 were discarded. The annotations were done through Mechanical Turk. Each annotator had to mark the positive, negative, and neutral parts of a message as well as to provide the overall polarity label for the message. Later, the annotations were combined through intersection for the term-level task and by majority voting for the message-level task. The details on data collection and annotation were released to the participants after the competition.

The data characteristics for both tasks are shown in Table 1. The training set was distributed through tweet ids and a download script. However, not all tweets were accessible. For example, a Twitter user could have deleted her messages, and thus these messages would not be available. Table 1 shows the number of the training examples we were able to download. The development and test sets were provided in full by FTP.

The SemEval-2013 training and development data are used to train our supervised sentiment analysis system presented in Section 5. The performance of the system is evaluated on both test sets, tweets and SMS (Section 6.2). The test data are also used in the experiments on comparing the performance of sentiment lexicons in unsupervised settings (Section 6.1).

In addition, we evaluate the system on the dataset of movie review excerpts. The task is to predict the sentiment label (positive or negative) of a given sentence, extracted from a longer movie review (message-level task). The original data were collected by Pang and Lee

(2005). In this work, we use the dataset and the evaluation setup provided by Socher et al. (2013). The dataset is comprised of 4,963 positive and 4,650 negative sentences split into the training (6,920 sentences), development (872 sentences), and test (1,821 sentences) sets. Since detailed phrase-level annotations are not available for most real-world applications, we use only sentence-level annotations and ignore the phrase-level annotations and the parse-tree structures of the sentences provided with the data. We train our sentiment analysis system on the training and development subsets and evaluate its performance on the test subset. The results of these experiments are reported in Section 6.2.

4. Sentiment Lexicons Used by Our System

4.1 Existing, General-Purpose, Manually Created Sentiment Lexicons

Most of the lexicons that were created by manual annotation tend to be domain free and include a few thousand terms. The lexicons that we use include the NRC Emotion Lexicon (Mohammad & Turney, 2010), Bing Liu's Lexicon (Hu & Liu, 2004), and the MPQA Subjectivity Lexicon (Wilson et al., 2005). The NRC Emotion Lexicon is comprised of frequent English nouns, verbs, adjectives, and adverbs annotated for eight emotions (joy, sadness, anger, fear, disgust, surprise, trust, and anticipation) as well as for positive and negative sentiment. Bing Liu's Lexicon provides a list of positive and negative words manually extracted from customer reviews. The MPQA Subjectivity Lexicon contains words marked with their prior polarity (positive or negative) and the strength of evaluative intensity (strong or weak). Entities in these lexicons do not come with a real-valued score indicating the fine-grained evaluative intensity.

4.2 New, Tweet-Specific, Automatically Generated Sentiment Lexicons

4.2.1 Hashtag Sentiment Lexicon

Certain words in tweets are specially marked with a hashtag (#) and can indicate the topic or sentiment. Mohammad (2012) showed that hashtagged emotion words such as joy, sadness, angry, and surprised are good indicators that the tweet as a whole (even without the hashtagged emotion word) is expressing the same emotion. We adapted that idea to create a large corpus of positive and negative tweets. From this corpus we then automatically generated a high-coverage, tweet-specific sentiment lexicon as described below.

We polled the Twitter API every four hours from April to December 2012 in search of tweets with either a positive-word hashtag or a negative-word hashtag. A collection of 77 seed words closely associated with positive and negative sentiment such as *#good, #excellent, #bad,* and *#terrible* were used (30 positive and 47 negative). These terms were chosen from entries for *positive* and *negative* in Roget's Thesaurus[5]. About 2 million tweets were collected in total. We used the metadata tag "iso_language_code" to identify English tweets. Since this tag is not always reliable, we additionally discarded tweets that did not

5. http://www.gutenberg.org/ebooks/10681

have at least two valid English content words from Roget's Thesaurus.[6] This step also helped discard very short tweets and tweets with a large proportion of misspelled words.

A set of 775,000 remaining tweets, which we refer to as *Hashtag Sentiment Corpus*, was used to generate a large word–sentiment association lexicon. A tweet was considered positive if it had one of the 30 positive hashtagged seed words, and negative if it had one of the 47 negative hashtagged seed words. The sentiment score for a term w was calculated from these pseudo-labeled tweets as shown below:

$$Sentiment\ Score\,(w) = PMI\,(w, positive) - PMI\,(w, negative) \tag{1}$$

PMI stands for pointwise mutual information:

$$PMI\,(w, positive) = log_2\ \frac{freq\,(w, positive) * N}{freq\,(w) * freq\,(positive)} \tag{2}$$

where *freq*(w, *positive*) is the number of times a term w occurs in positive tweets, *freq*(w) is the total frequency of term w in the corpus, *freq*(*positive*) is the total number of positive tweets, and N is the total number of tokens in the corpus. *PMI*(w, *negative*) is calculated in a similar way. Thus, equation 1 is simplified to:

$$Sentiment\ Score\,(w) = log_2\ \frac{freq\,(w, positive) * freq\,(negative)}{freq\,(w, negative) * freq\,(positive)} \tag{3}$$

Since PMI is known to be a poor estimator of association for low-frequency events, we ignore terms that occurred less than five times in each (positive and negative) group of tweets.

A positive sentiment score indicates a greater overall association with positive sentiment, whereas a negative score indicates a greater association with negative sentiment. The magnitude is indicative of the degree of association. The final lexicon, which we will refer to as *Hashtag Sentiment Base Lexicon (HS Base)* has entries for 39,413 unigrams and 178,851 bigrams. Entries were also generated for unigram–unigram, unigram–bigram, and bigram–bigram pairs that were not necessarily contiguous in the tweets corpus. Pairs with certain punctuations, '@' symbol, and some function words were removed. The lexicon has entries for 308,808 non-contiguous pairs.

4.2.2 Sentiment140 Lexicon

The *Sentiment140 Corpus* (Go et al., 2009) is a collection of 1.6 million tweets that contain emoticons. The tweets are labeled positive or negative according to the emoticon. We generated the *Sentiment140 Base Lexicon (S140 Base)* from this corpus in the same manner as described above for the hashtagged tweets using Equation 1. This lexicon has entries for 65,361 unigrams, 266,510 bigrams, and 480,010 non-contiguous pairs. In the following section, we further build on the proposed approach to create separate lexicons for terms in affirmative contexts and for terms in negated contexts.

6. Any word in the thesaurus was considered a content word with the exception of the words from the stopword list built by Gerard Salton and Chris Buckley for the SMART information retrieval system at Cornell University (http://www.lextek.com/manuals/onix/stopwords2.html).

4.3 Affirmative Context and Negated Context Lexicons

A word in a negated context has a different evaluative nature than the same word in an affirmative (non-negated) context. This difference may include the change in the polarity category (positive becomes negative or vice versa), the evaluative intensity, or both. For example, highly positive words (e.g., *great*) when negated tend to experience both, polarity change and intensity decrease, forming mildly negative phrases (e.g., *not great*). On the other hand, many strong negative words (e.g., *terrible*) when negated keep their negative polarity and just shift their intensity. The conventional approach of reversing polarity is not able to handle these cases properly.

We propose an empirical method to determine the sentiment of words in the presence of negation. We create separate lexicons for affirmative and negated contexts. In this way, two sentiment scores for each term w are computed: one for affirmative contexts and another for negated contexts. The lexicons are created as follows. The Hashtag Sentiment Corpus is split into two parts: *Affirmative Context Corpus* and *Negated Context Corpus*. Following the work by Pang, Lee, and Vaithyanathan (2002), we define a negated context as a segment of a tweet that starts with a negation word (e.g., *no, shouldn't*) and ends with one of the punctuation marks: ',', '.', ':', ';', '!', '?'. The list of negation words was adopted from Christopher Potts' sentiment tutorial.[7] Thus, part of a tweet that is marked as negated is included into the Negated Context Corpus while the rest of the tweet becomes part of the Affirmative Context Corpus. The sentiment label for the tweet is kept unchanged in both corpora. Then, we generate the *Affirmative Context Lexicon (HS AffLex)* from the Affirmative Context Corpus and the *Negated Context Lexicon (HS NegLex)* from the Negated Context Corpus using the technique described in Section 4.2. We will refer to the sentiment score calculated from the Affirmative Context Corpus as $score_{AffLex}(w)$ and the score calculated from the Negated Context Corpus as $score_{NegLex}(w)$. Similarly, the *Sentiment140 Affirmative Context Lexicon (S140 AffLex)* and the *Sentiment140 Negated Context Lexicon (S140 NegLex)* are built from the Affirmative Context and the Negated Context parts of the Sentiment140 tweet corpus. To employ these lexicons on a separate dataset, we apply the same technique to split each message into affirmative and negated contexts and then match words in affirmative contexts against the Affirmative Context Lexicons and words in negated contexts against the Negated Context Lexicons.

Computing a sentiment score for a term w only from affirmative contexts makes $score_{AffLex}(w)$ more precise since it is no longer polluted by negation. Positive terms get stronger positive scores and negative terms get stronger negative scores. Furthermore, for the first time, we create lexicons for negated terms and compute $score_{NegLex}(w)$ that reflects the behaviour of words in the presence of negation. Table 2 shows a few examples of positive and negative terms with their sentiment scores from the Sentiment140 Base, Affirmative Context (AffLex) and Negated Context (NegLex) Lexicons. In Fig. 1, we visualize the relationship between $score_{AffLex}(w)$ and $score_{NegLex}(w)$ for a set of words manually annotated for sentiment in the MPQA Subjectivity Lexicon. The x-axis is $score_{AffLex}(w)$, the sentiment score of a term w in the Sentiment140 Affirmative Context Lexicon; the y-axis is $score_{NegLex}(w)$, the sentiment score of a term w in the Sentiment140 Negated Context Lexicon. Dots in the plot correspond to words that occur in each of the MPQA Subjectiv-

7. http://sentiment.christopherpotts.net/lingstruc.html

Table 2: Example sentiment scores from the Sentiment140 Base, Affirmative Context (AffLex) and Negated Context (NegLex) Lexicons.

Term	**Sentiment140 Lexicons**		
	Base	**AffLex**	**NegLex**
Positive terms			
great	1.177	1.273	-0.367
beautiful	1.049	1.112	0.217
nice	0.974	1.149	-0.912
good	0.825	1.167	-1.414
honest	0.391	0.431	-0.123
Negative terms			
terrible	-1.766	-1.850	-0.890
shame	-1.457	-1.548	-0.722
bad	-1.297	-1.674	0.021
ugly	-0.899	-0.964	-0.772
negative	-0.090	-0.261	0.389

ity Lexicon, the Sentiment140 Affirmative Context Lexicon, and the Sentiment140 Negated Context Lexicon. Furthermore, we discard the terms whose polarity category (positive or negative) in the Sentiment140 Affirmative Context Lexicon does not match their polarity in the MPQA Subjectivity Lexicon. We observe that when negated, 76% of the positive terms reverse their polarity whereas 82% of the negative terms keep their polarity orientation and just shift their sentiment scores. (This behaviour agrees well with human judgments from the study by Taboada et al. (2011).) Changes in evaluative intensity vary from term to term. For example, $score_{NegLex}(good) < -score_{AffLex}(good)$ whereas $score_{NegLex}(great) > -score_{AffLex}(great)$.

We also compiled a list of 596 antonym pairs from WordNet and compare the scores of terms in the Sentiment140 Affirmative Context Lexicon with the scores of the terms' antonyms in the Sentiment140 Negated Context Lexicon. We found that 51% of negated positive terms are less negative than their corresponding antonyms (e.g., $score_{NegLex}(good) > score_{AffLex}(bad)$), but 95% of negated negative terms are more negative than their positive antonyms (e.g., $score_{NegLex}(ugly) < score_{AffLex}(beautiful)$).

These experiments reveal the tendency of positive terms when negated to convey a negative sentiment and the tendency of negative terms when negated to still convey a negative sentiment. Moreover, the degree of change in evaluative intensity appears to be term-dependent. Capturing all these different behaviours of terms in negated contexts by means of the Negated Context Lexicons empower our automatic sentiment analysis system as we demonstrate through experiments in Section 6. Furthermore, we believe that the Affirmative Context Lexicons and the Negated Context Lexicons can be valuable in other applications such as textual entailment recognition and paraphrase detection. For instance in the paraphrase detection task, given two sentences *"The hotel room wasn't terrible."* and *"The hotel room was excellent."* an automatic system can correctly infer that these sentences are

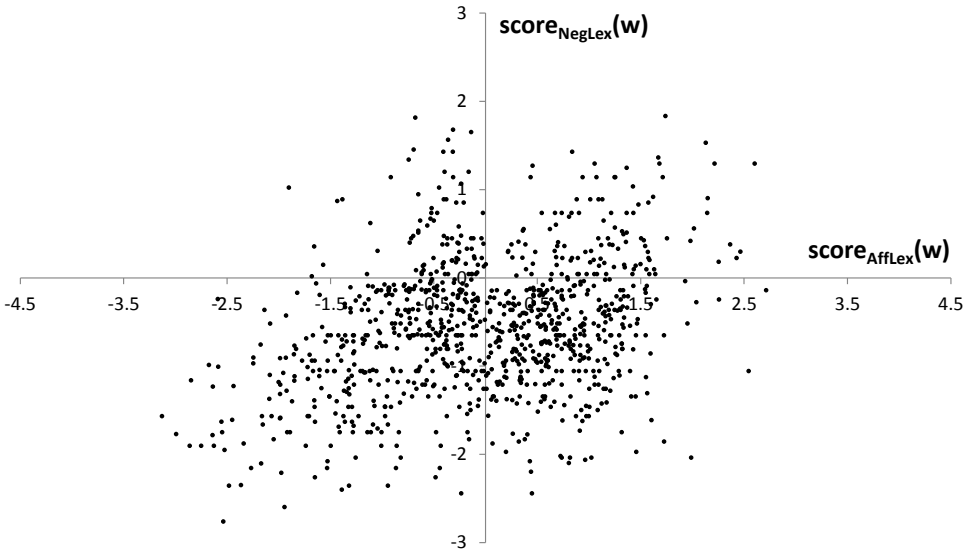

Figure 1: The sentiment scores from the Sentiment140 AffLex and the Sentiment140 NegLex for 480 positive and 486 negative terms from the MPQA Subjectivity Lexicon. The x-axis is $score_{AffLex}(w)$, the sentiment score of a term w in the Sentiment140 Affirmative Context Lexicon; the y-axis is $score_{NegLex}(w)$, the sentiment score of a term w in the Sentiment140 Negated Context Lexicon. Each dot corresponds to one (positive or negative) term. The graph shows that positive and negative terms when negated tend to convey a negative sentiment. Negation affects sentiment differently for each term.

not paraphrases by looking up $score_{NegLex}(terrible)$ and $score_{AffLex}(excellent)$ and seeing that the polarities and intensities of these terms do not match (i.e., $score_{AffLex}(excellent)$ is highly positive and $score_{NegLex}(terrible)$ is slightly negative). At the same time, a mistake can easily be made with conventional lexicons and the polarity reversing strategy, according to which the strong negative term *terrible* is assumed to convey a strong positive sentiment in the presence of negation and, therefore, the polarities and intensities of the two terms would match.

4.4 Negated Context (Positional) Lexicons

We propose to further improve the method of constructing the Negated Context Lexicons by splitting a negated context into two parts: the *immediate context* consisting of a *single* token that directly follows a negation word, and the *distant context* consisting of the rest of the tokens in the negated context. We refer to these lexicons as *Negated Context (Positional) Lexicons*. Each token in a Negated Context (Positional) Lexicon can have two scores: *immediate-context score* and *distant-context score*. The benefits of this approach are two-fold. Intuitively, negation affects words *directly* following a negation word more strongly than the words farther away. Compare, for example, immediate negation in *not good* and more distant negation in *not very good*, *not as good*, *not such a good idea*. Second,

Table 3: The number of positive and negative entries in the sentiment lexicons.

Lexicon	Positive	Negative	Total
NRC Emotion Lexicon	2,312 (41%)	3,324 (59%)	5,636
Bing Liu's Lexicon	2,006 (30%)	4,783 (70%)	6,789
MPQA Subjectivity Lexicon	2,718 (36%)	4,911 (64%)	7,629
Hashtag Sentiment Lexicons (HS)			
HS Base Lexicon			
- unigrams	19,121 (49%)	20,292 (51%)	39,413
- bigrams	69,337 (39%)	109,514 (61%)	178,851
HS AffLex			
- unigrams	19,344 (51%)	18,905 (49%)	38,249
- bigrams	67,070 (42%)	90,788 (58%)	157,858
HS NegLex			
- unigrams	936 (14%)	5,536 (86%)	6,472
- bigrams	3,954 (15%)	22,258 (85%)	26,212
Sentiment140 Lexicons (S140)			
S140 Base Lexicon			
- unigrams	39,979 (61%)	25,382 (39%)	65,361
- bigrams	135,280 (51%)	131,230 (49%)	266,510
S140 AffLex			
- unigrams	40,422 (63%)	23,382 (37%)	63,804
- bigrams	133,242 (55%)	107,206 (45%)	240,448
S140 NegLex			
- unigrams	1,038 (12%)	7,315 (88%)	8,353
- bigrams	5,913 (16%)	32,128 (84%)	38,041

immediate-context scores are less noisy. Our simple negation scope identification algorithm can occasionally fail and include into negated context parts of a tweet that are not actually negated (e.g., if a punctuation mark is missing). These errors have less effect on immediate context. When employing these lexicons, we use an immediate-context score for a word immediately preceded by a negation word and use distant-context scores for all other words affected by a negation. As before, for non-negated parts of a message, sentiment scores from an Affirmative Context Lexicon are used. Assuming that words occur in distant contexts more often than in immediate contexts, this approach can introduce more sparseness to the lexicons. Thus, we apply a back-off strategy: if an immediate-context score is not available for a token immediately following a negation word, its distant-context score is used instead. In Section 6, we experimentally show that the Negated Context (Positional) Lexicons provide additional benefits to our sentiment analysis system over the regular Negated Context Lexicons described in the previous section.

4.5 Lexicon Coverage

Table 3 shows the number of positive and negative entries in each of the sentiment lexicons discussed above. The automatically generated lexicons are an order of magnitude larger than the manually created lexicons. We can see that all manual lexicons contain more negative terms than positive terms. In the automatically generated lexicons, this imbalance is less pronounced (49% positive vs. 51% negative in the Hashtag Sentiment Base Lexicon)

Table 4: Lexicon's supplemental coverage: for row X and column Y, the number of Lexicon X's entries that are not found in Lexicon Y and (in brackets) the percentage of tokens in the SemEval-2013 tweet test set covered by these extra entries of Lexicon X. 'NRC' stands for NRC Emotion Lexicon, 'B.L.' is for Bing Liu's Lexicon, 'MPQA' is for MPQA Subjectivity Lexicon, 'HS' is for Hashtag Sentiment Base Lexicon, 'S140' is for Sentiment140 Base Lexicon.

Lexicon	**NRC**	**B.L.**	**MPQA**	**HS**	**S140**
NRC	-	3,179 (2.25%)	3,010 (2.00%)	2,480 (0.09%)	1,973 (0.05%)
B.L.	4,410 (1.72%)	-	1,383 (0.70%)	4,001 (0.07%)	3,457 (0.05%)
MPQA	3,905 (3.37%)	1,047 (2.60%)	-	3,719 (0.07%)	3,232 (0.04%)
HS	36,338 (64.23%)	36,628 (64.73%)	36,682 (62.84%)	-	15,185 (0.59%)
S140	61,779 (64.13%)	62,032 (64.65%)	62,143 (62.74%)	41,133 (0.53%)	-

or even reversed (61% positive vs. 39% negative in the Sentiment140 Base Lexicon). The Sentiment140 Base Lexicon was created from an equal number of positive and negative tweets. Therefore, the prevalence of positive terms corresponds to the general trend in language and supports the Polyanna Hypothesis (Boucher & Osgood, 1969), which states that people tend to use positive terms more frequently and diversely than negative. Note, however, that negative terms are dominant in the Negated Context Lexicons since most terms, both positive and negative, tend to convey negative sentiment in the presence of negation. The overall sizes of the Negated Context Lexicons are rather small since negation occurs only in 24% of the tweets in the Hashtag and Sentiment140 corpora and only part of a message with negation is actually negated.

Table 4 shows the differences in coverage between the lexicons. Specifically, it gives the number of additional terms a lexicon in row X has in comparison to a lexicon in column Y and the percentage of tokens in the SemEval-2013 tweet test set covered by these extra entries of lexicon X (numbers in brackets). For instance, almost half of Bing Liu's Lexicon (3,457 terms) is not found in the Sentiment140 Base Lexicon. However, these additional terms represent only 0.05% of all the tokens from the tweet test set. These are terms that are rarely used in short informal writing (e.g., *acrimoniously*, *bestial*, *nepotism*). Each of the manually created lexicons covers extra 2–3% of the test data compared to other manual lexicons. On the other hand, the automatically generated lexicons cover 60% more tokens in the test data. Both automatic lexicons provide a number of terms not found in the other.

5. Our System

5.1 Classifier

Our system, NRC-Canada Sentiment Analysis System, employs supervised statistical machine learning. For both tasks, message-level and term-level, we learn a linear-kernel Support Vector Machine (SVM) (Chang & Lin, 2011) classification model from the training data. SVM is a state-of-the-art learning algorithm proved to be effective on text categorization tasks and robust on large feature spaces. In the preliminary experiments, a linear-kernel

SVM outperformed a maximum-entropy classifier. Also, a linear-kernel SVM showed better performance than SVM with other kernels implemented in LibSVM.

The classification model leverages a variety of surface-form, semantic, and sentiment lexicon features described below. The sentiment lexicon features are derived from three existing, general-purpose, manual lexicons (NRC Emotion Lexicon, Bing Liu's Lexicon, and MPQA Subjectivity Lexicon), and four newly created, tweet-specific lexicons (Hashtag Sentiment Affirmative Context, Hashtag Sentiment Negated Context (Positional), Sentiment140 Affirmative Context, and Sentiment140 Negated Context (Positional)).

5.2 Features

5.2.1 Message-Level Task

For the message-level task, the following pre-processing steps are performed. URLs and userids are normalized to http://someurl and @someuser, respectively. Tweets are tokenized and part-of-speech tagged with the Carnegie Mellon University (CMU) Twitter NLP tool (Gimpel, Schneider, O'Connor, Das, Mills, Eisenstein, Heilman, Yogatama, Flanigan, & Smith, 2011). Then, each tweet is represented as a feature vector made up of the following groups of features:

- word ngrams: presence or absence of contiguous sequences of 1, 2, 3, and 4 tokens; non-contiguous ngrams (ngrams with one token replaced by *);
- character ngrams: presence or absence of contiguous sequences of 3, 4, and 5 characters;
- all-caps: the number of tokens with all characters in upper case;
- POS: the number of occurrences of each part-of-speech tag;
- hashtags: the number of hashtags;
- negation: the number of negated contexts. Negation also affects the ngram features: a word w becomes w_NEG in a negated context;
- sentiment lexicons:
  - **Automatic lexicons** The following sets of features are generated separately for the Hashtag Sentiment Lexicons (HS AffLex and HS NegLex (Positional)) and the Sentiment140 Lexicons (S140 AffLex and S140 NegLex (Positional)). For each token w occurring in a tweet and present in the lexicons, we use its sentiment score ($score_{AffLex}(w)$ if w occurs in an affirmative context and $score_{NegLex}(w)$ if w occurs in a negated context) to compute:
    - the number of tokens with $score(w) \neq 0$;
    - the total score = $\sum_{w \in tweet} score(w)$;
    - the maximal score = $max_{w \in tweet} score(w)$;
    - the score of the last token in the tweet.

These features are calculated for all positive tokens (tokens with sentiment scores greater than zero), for all negative tokens (tokens with sentiment scores less than zero), and for all tokens in a tweet. Similar feature sets are also created for each part-of-speech tag and for hashtags. Separate feature sets are produced for unigrams, bigrams, and non-contiguous pairs.

- **Manual lexicons** For each of the three manual sentiment lexicons (NRC Emotion Lexicon, Bing Liu's Lexicon, and MPQA Subjectivity Lexicon), we compute the following four features:
  - the sum of positive scores for tweet tokens in affirmative contexts;
  - the sum of negative scores for tweet tokens in affirmative contexts;
  - the sum of positive scores for tweet tokens in negated contexts;
  - the sum of negative scores for tweet tokens in negated contexts.

  Negated contexts are identified exactly as described earlier in Section 4.3 (the method for creating the Negated Context Corpora). The remaining parts of the messages are treated as affirmative contexts. We use the score of +1 for positive entries and the score of -1 for negative entries for the NRC Emotion Lexicon and Bing Liu's Lexicon. For MPQA Subjectivity Lexicon, which provides two grades of the association strength (strong and weak), we use scores +1/-1 for weak associations and +2/-2 for strong associations. The same feature sets are also created for each part-of-speech tag, for hashtags, and for all-caps tokens.

- punctuation:
  - the number of contiguous sequences of exclamation marks, question marks, and both exclamation and question marks;
  - whether the last token contains an exclamation or question mark;

- emoticons: The polarity of an emoticon is determined with a regular expression adopted from Christopher Potts' tokenizing script:[8]
  - presence or absence of positive and negative emoticons at any position in the tweet;
  - whether the last token is a positive or negative emoticon;

- elongated words: the number of words with one character repeated more than two times, for example, *soooo*;

- clusters: The CMU Twitter NLP tool provides token clusters produced with the Brown clustering algorithm on 56 million English-language tweets. These 1,000 clusters serve as alternative representation of tweet content, reducing the sparcity of the token space.
  - the presence or absence of tokens from each of the 1000 clusters.

8. http://sentiment.christopherpotts.net/tokenizing.html

5.2.2 Term-level Task

The pre-processing steps for the term-level task include tokenization and stemming with Porter stemmer (Porter, 1980).[9] Then, each tweet is represented as a feature vector with the following groups of features:

- word ngrams:
  - presence or absence of unigrams, bigrams, and the full word string of a target term;
  - leading and ending unigrams and bigrams;
- character ngrams: presence or absence of two- and three-character prefixes and suffixes of all the words in a target term (note that the target term may be a multi-word sequence);
- upper case:
  - whether all the words in the target start with an upper case letter followed by lower case letters;
  - whether the target words are all in uppercase (to capture a potential named entity);
- stopwords: whether a term contains only stop-words. If so, a separate set of features indicates whether there are 1, 2, 3, or more stop-words;
- negation: similar to the message-level task;
- sentiment lexicons: for each of the manual sentiment lexicons (NRC Emotion Lexicon, Bing Liu's Lexicon, and MPQA Subjectivity Lexicon) and automatic sentiment lexicons (HS AffLex and HS NegLex (Positional), and S140 AffLex and S140 NegLex (Positional) Lexicons), we compute the following three features:
  - the sum of positive scores;
  - the sum of negative scores;
  - the total score.

  For the manual lexicons, the polarity reversing strategy is applied to negation.[10] Note that words themselves and not their stems are matched against the sentiment lexicons.
- punctuation: presence or absence of punctuation sequences such as '?!' and '!!!';
- emoticons: the numbers and categories of emoticons that a term contains[11];
- elongated words: presence or absence of elongated words;

9. Some differences in implementation, such as the use of a stemmer, are simply a result of different team members working on the two tasks.
10. In the experiments on the development dataset, these manual lexicon features showed better performance on the term-level task than the set of four features used for the message-level task.
11. http://en.wikipedia.org/wiki/List_of_emoticons

- lengths:
  - the length of a target term (number of words);
  - the average length of words (number of characters) in a term;
  - a binary feature indicating whether a term contains long words;
- position: whether a term is at the beginning, at the end, or at another position in a tweet;
- term splitting: when a term contains a hashtag made of multiple words (e.g., *#biggestdaythisyear*), we split the hashtag into component words;
- others:
  - whether a term contains a Twitter user name;
  - whether a term contains a URL.

The above features are extracted from target terms as well as from the rest of the message (the context). For unigrams and bigrams, we use four words on either side of the target as the context. The window size was chosen through experiments on the development set.

6. Experiments

Several experiments on the sentiment analysis of short informal texts are carried out to demonstrate the superior predictive power of the new, tweet-specific, automatically created lexicons over existing, general-purpose lexicons. Further, we show that the Negated Context Lexicons can bring additional gains over the standard polarity reversing strategy of handling negation.

We begin with the experiments in unsupervised settings (Section 6.1). Our goal is to compare the predictive capacity of the lexicons with the simplest setup to reduce the influence of other factors (such as the choice of features) as much as possible. Also, we evaluate the impact of the amount of data used to create an automatic lexicon on the quality of the lexicon. Then, in Section 6.2 we analyze the contributions of features derived from different sentiment lexicons to our supervised sentiment analysis system.

6.1 Lexicon Performance with No Supervision

In the first set of experiments, we evaluate the performance of each individual lexicon on the message-level task in unsupervised settings. No training and/or tuning is performed. Since most of the lexicons provide the association scores for the positive and negative classes only, in this subsection, we reduce the problem to a two-way classification task (positive or negative). The SemEval-2013 tweet test set and SMS test set are used for evaluation. The neutral instances are removed from both datasets.

To classify a message as positive or negative, we add up the scores for all matches in a particular lexicon and assign a positive label if the cumulative score is greater than zero and a negative label if the cumulative score is less than zero. Again, we use scores +1/-1 for

the NRC Emotion Lexicon and Bing Liu's Lexicon and scores +1/-1 for weak associations and +2/-2 for strong associations in the MPQA Subjectivity Lexicon. A message is left unclassified when the score is equal to zero or when no matches are found.

Table 5 presents the results of unsupervised polarity detection for (1) manually created, general-purpose lexicons (NRC Emotion Lexicon, Bing Liu's Lexicon, and MPQA Subjectivity Lexicon), (2) automatically created, general-purpose lexicons (SentiWordNet 3.0 (Baccianella, Esuli, & Sebastiani, 2010), MSOL (Mohammad et al., 2009), and Osgood Evaluative Factor Ratings (Turney & Littman, 2003)), and (3) our automatically created, tweet-specific lexicons (Hashtag Sentiment and Sentiment140 Lexicons). Only unigram entries are used from each lexicon. The automatic general-purpose lexicons are large, open-domain lexicons providing automatically generated sentiment scores for words taken from hand-built general thesauri such as WordNet and Macquarie Thesaurus.[12] The predictive performance is assessed through precision and recall on the positive and negative classes as well as the macro-averaged F-score of the two classes. Observe that for most of the lexicons, both precision and recall on the negative class are lower than on the positive class. In particular, this holds for all the manual lexicons (rows a–c) despite the fact that they have significantly more negative terms than positive terms. One possible explanation for this phenomenon is that people can express negative sentiment without using many or any clearly negative words.

The threshold of zero seems natural for separating the positive and negative classes in unsupervised polarity detection; however, better results are possible with other thresholds. For example, predictions produced by the Osgood Evaluative Factor Ratings (rows f) are highly skewed towards the positive class (recall of 95.42 on the positive class and 31.28 on the negative class), which negatively affects its macro-averaged F-score. To avoid the problem of setting the optimal threshold in unsupervised settings, we report the Area Under the ROC curve (AUC), which takes into account the performance of the classifier at all possible thresholds (see the last column in Table 5). To calculate AUC, the cumulative scores assigned by a lexicon to the test messages are ordered in the decreasing order. Then, taking every score as a possible threshold, the true positive ratio is plotted against the false positive ratio and the area under this curve is calculated. It has been shown that the AUC of a classifier is equivalent to the probability that the classifier will rank a randomly chosen positive instance higher than a randomly chosen negative instance. This is also equivalent to the Wilcoxon test of ranks (Hanley & McNeil, 1982).

All automatically generated lexicons match at least one token in each test message while the manual lexicons are unable to cover 10–20% of the tweet test set. Paying attention to negation proves important for all general-purpose lexicons: both the macro-averaged F-score and AUC are improved by 1–4 percentage points. However, this is not the case for the Hashtag Sentiment Base (rows g) and the Sentiment140 Base Lexicons (rows k). The polarity reversing strategy fails to improve over the simple baseline of disregarding negation on these lexicons.

Compared to the Base Lexicons, the lexicons created only from affirmative contexts (rows h and l) are more precise and slightly improve the predictive performance. More substantial improvements are obtained by adding the Negated Context Lexicons (rows i and

12. The SentiWordNet 3.0 has 30,821 unigrams, the MSOL Lexicon has 55,141 unigrams, and the Osgood Evaluative Factor Ratings Lexicon contains ratings for 72,905 unigrams.

Table 5: Prediction performance of the unigram lexicons in unsupervised polarity detection on the SemEval-2013 tweet test set. 'Cover.' denotes coverage – the percentage of tweets in the test set with at least one match from the lexicon; P is precision; R is recall; F_{avg} is the macro-averaged F-score for the positive and negative classes; AUC is the area under the ROC curve.

Lexicon	**Cover.**	**Positive**		**Negative**		F_{avg}	**AUC**
		P	**R**	**P**	**R**		
Manual general-purpose lexicons							
a. NRC Emotion Lexicon							
- disregarding negation	76.30	84.77	58.78	56.83	34.61	56.22	70.66
- reversing polarity	76.30	86.20	59.61	59.02	35.94	57.58	72.83
b. Bing Liu's Lexicon							
- disregarding negation	77.59	90.73	61.64	65.94	45.42	63.60	79.08
- reversing polarity	77.59	92.02	61.64	66.74	48.75	65.09	80.20
c. MPQA Subjectivity Lexicon							
- disregarding negation	88.36	82.90	71.56	58.57	38.10	61.49	73.01
- reversing polarity	88.36	84.56	71.06	60.09	43.09	63.71	75.33
Automatic general-purpose lexicons							
d. SentiWordNet 3.0							
- disregarding negation	100.00	82.40	71.76	44.93	59.73	64.00	71.51
- reversing polarity	100.00	85.08	71.12	47.42	67.22	66.54	75.15
e. MSOL							
- disregarding negation	100.00	77.18	74.43	38.66	27.79	54.06	63.44
- reversing polarity	100.00	77.35	74.30	41.70	30.95	55.66	63.80
f. Osgood Evaluative Factor Ratings							
- disregarding negation	100.00	75.65	97.65	74.31	17.80	56.99	75.30
- reversing polarity	100.00	78.41	95.42	72.31	31.28	64.88	80.11
Automatic tweet-specific lexicons							
g. HS Base Lexicon							
- disregarding negation	100.00	89.15	72.65	51.79	76.87	70.97	82.52
- reversing polarity	100.00	88.03	72.07	50.45	74.38	69.69	80.21
h. HS AffLex							
- disregarding negation	100.00	87.53	80.41	57.75	70.05	73.56	83.06
- reversing polarity	100.00	87.04	79.07	55.84	69.22	72.34	82.21
i. HS AffLex and HS NegLex	100.00	89.44	77.04	55.92	76.21	73.64	84.61
j. HS AffLex and HS NegLex (Posit.)	100.00	89.60	77.29	56.30	76.54	73.94	84.62
k. S140 Base Lexicon							
- disregarding negation	100.00	88.60	77.61	55.78	73.88	73.15	84.47
- reversing polarity	100.00	87.78	77.23	54.68	71.88	72.14	83.21
l. S140 AffLex							
- disregarding negation	100.00	85.96	86.45	64.02	63.06	74.87	84.94
- reversing polarity	100.00	87.19	85.31	63.56	67.05	75.75	86.04
m. S140 AffLex and S140 NegLex	100.00	89.65	83.21	63.03	74.88	77.37	86.88
n. S140 AffLex and S140 NegLex (Posit.)	100.00	89.79	83.33	63.31	75.21	77.59	87.14

Table 6: Prediction performance of the unigram lexicons in unsupervised polarity detection on the SemEval-2013 SMS test set. The polarity reversing strategy is applied to negation for all lexicons except for the Negated Context Lexicons. 'Cover.' denotes coverage – the percentage of SMS in the test set with at least one match from the lexicon; P is precision; R is recall; F_{avg} is the macro-averaged F-score for the positive and negative classes; AUC is the area under the ROC curve.

Lexicon	**Cover.**	**Positive**		**Negative**		F_{avg}	**AUC**
		P	**R**	**P**	**R**		
Manual general-purpose lexicons							
a. NRC Emotion Lexicon	70.88	85.11	56.91	80.17	47.21	63.82	79.66
b. Bing Liu's Lexicon	69.75	87.90	61.99	86.36	48.22	67.30	83.24
c. MPQA Subjectivity Lexicon	83.86	81.69	72.56	77.95	52.03	69.63	82.42
Automatic general-purpose lexicons							
d. SentiWordNet 3.0	100.00	77.36	79.88	73.87	70.30	75.32	81.34
e. MSOL	100.00	69.88	73.58	69.14	44.92	63.07	72.49
f. Osgood Evaluative Factor Ratings	100.00	66.15	95.33	87.01	39.09	66.02	84.01
Automatic tweet-specific lexicons							
g. HS Base Lexicon	100.00	88.41	41.87	56.20	93.15	63.47	75.49
i. HS AffLex and HS NegLex	100.00	92.03	46.95	58.90	94.92	67.44	81.67
j. HS AffLex and HS NegLex (Posit.)	100.00	92.00	46.75	58.81	94.92	67.31	82.05
k. S140 Base Lexicon	100.00	85.71	73.17	71.67	84.77	78.31	86.07
m. S140 AffLex and S140 NegLex	100.00	88.38	78.86	76.73	87.06	82.46	89.34
n. S140 AffLex and S140 NegLex (Posit.)	100.00	88.69	79.67	77.48	87.31	83.02	89.60

m). Furthermore, the Sentiment140 Negated Context (Positional) Lexicon (row n) offers additional gain of 0.26 percentage points in AUC over the regular Sentiment140 Negated Context Lexicon (row m). Overall, the Affirmative Context Lexicons and the Negated Context (Positional) Lexicons outperform the Base Lexicons by over 2 percentage points in AUC.

The automatically created general-purpose lexicons (rows d–f) have a substantially higher coverage; however, they do not show better performance than the manual lexicons. On the other hand, all our tweet-specific automatic lexicons demonstrate a predictive power superior to that of both, the manually and automatically created, general-purpose lexicons. The differences are especially pronounced for the Affirmative Context Lexicons and the Negated Context Lexicons. While keeping the level of precision close to that of the manual lexicons, the automatic tweet-specific lexicons are able to substantially improve the recall on both positive and negative classes. This increase in recall is particularly noticeable on the negative class where the differences reach forty percentage points.

Similar trends can be observed on the SMS test set (see Table 6). The automatic lexicons built separately for affirmative and negated contexts (rows i and m) perform 3–6 percentage points better than the corresponding Base Lexicons in combination with the polarity reversing strategy (rows g and k). Moreover, the use of the Sentiment140 Affirmative Context Lexicon and Negated Context (Positional) Lexicon (row n) again results in higher

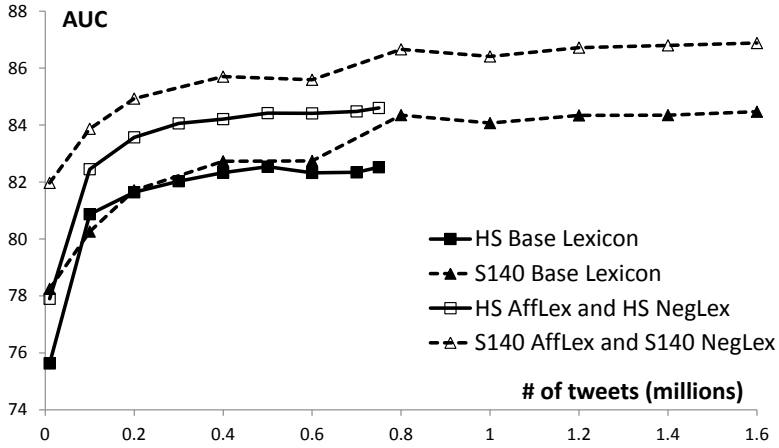

Figure 2: Performance of the automatic tweet-specific lexicons in unsupervised polarity detection on the SemEval-2013 tweet test set for different sizes of the tweet corpora. "AUC" denotes the Area Under the ROC Curve.

performance than that obtained with any other manually or automatically created lexicon we used.

To get a better understanding of the impact of the amount of data used to create an automatic lexicon on the quality of the lexicon, we compare the performance of the automatic lexicons built from subsets of the available data. We split a tweet corpus (Hashtag Sentiment Corpus or Sentiment140 Corpus) into smaller chunks by the tweets' time stamp. Fig. 2 shows the performance of the Hashtag Sentiment Base, Hashtag Sentiment Affirmative Context and Hashtag Sentiment Negated Context Lexicons, Sentiment140 Base, and Sentiment140 Affirmative Context and Sentiment140 Negated Context Lexicons built from these partial corpora as a function of the corpus' size. As above, the performance of the lexicons is evaluated in terms of AUC in unsupervised polarity detection on the SemEval-2013 tweet test set. We can see that the Sentiment140 Lexicons generated from half of the available tweet set still have higher predictive power than the full Hashtag Sentiment Lexicons. Interestingly, both Hashtag Sentiment Lexicons seem to stabilize at the corpus' size of 400,000–500,000 tweets whereas both Sentiment140 Lexicons stabilize at about 800,000 tweets. However, better results might still be possible with corpora that are orders of magnitude larger.

6.2 Lexicon Performance in Supervised Sentiment Analysis

In this section, we evaluate our supervised sentiment analysis system (described in Section 5) on a three-class problem (positive, negative, and neutral) on both the message-level task and the term-level task. We use the data provided for the SemEval-2013 competition. We examine the contribution of various feature groups, including the features derived from the sentiment lexicons: manually created lexicons (NRC Emotion Lexicon, Bing Liu's Lexicon, and MPQA Subjectivity Lexicon) and our automatically created lexicons (Hashtag

Table 7: Message-level task: The macro-averaged F-scores on the SemEval-2013 datasets.

Classifier	**Train. Set**	**Dev. Set**	**Test Sets**	
			Tweets	**SMS**
a. Majority baseline	26.94	26.85	29.19	19.03
b. SVM-unigrams	36.95	36.71	39.61	39.29
c. Our system:				
c.1. official SemEval-2013 submission	67.09	68.72	69.02	68.46
c.2. best result	68.19	68.43	70.45	69.77

Sentiment and Sentiment140 Lexicons). Finally, we compare the performance of different strategies to process negation.

For both tasks, we train an SVM classifier on the provided training data and evaluate the performance of the learned models on an unseen tweet test set. The same models are applied, without any change, to the test set of SMS messages. We evaluate the performance with the bottom-line evaluation measure used by all participants in the SemEval-2013: the macro-averaged F-score of the positive and negative classes. We report the results obtained by our system on the training set (ten-fold cross-validation), development set (when trained on the training set), and test sets (when trained on the combined set of tweets in the training and development sets). Significance tests are performed using a one-tailed paired t-test with approximate randomization at the $p < .05$ level (Yeh, 2000).

In order to test our system on a different domain, we conduct experiments on classifying movie review sentences as positive or negative (message-level task only). We use the dataset and the evaluation setup provided by Socher et al. (2013). We train the system on the training and development subsets of the movie review excerpts dataset and apply the learned model on the test subset. To compare with published results on this dataset, we use accuracy as the evaluation measure.

6.2.1 Results for the Message-Level Task

(a) On the SemEval-2013 data: The results obtained by our system on the SemEval-2013 message-level task are presented in Table 7. Our official submission on this task (row c.1) obtained a macro-averaged F-score of 69.02 on the tweet test set and 68.46 on the SMS test set. Out of 48 submissions from 34 teams, our system ranked first on both datasets.[13] After replacing the Base Lexicons with the Affirmative Context Lexicons and the Negated Context (Positional) Lexicons and with some improvements to the feature set, we achieved the scores of 70.45 on the tweet set and 69.77 on the SMS set (row c.2).[14] The differences between the best scores and the official scores on both test sets are statistically significant. The table also shows the baseline results obtained by a majority classifier that always predicts the most frequent class (row a). The bottom-line F-score is based only on the F-scores of the positive and negative classes (and not on neutral), so the majority baseline chooses the most frequent class among positive and negative, which in this case is

13. The second-best results were 65.27 on the tweet set and 62.15 on the SMS set.
14. The contributions of the different versions of the automatic lexicons to the overall system's performance are presented later in this subsection.

Table 8: Message-level task: The macro-averaged F-scores obtained on the SemEval-2013 datasets when one of the feature groups is removed. Scores marked with * are statistically significantly different ($p < .05$) from the corresponding scores in row a.

Experiment	**Train. Set**	**Dev. Set**	**Test Sets**	
			Tweets	**SMS**
a. all features	68.19	68.43	70.45	69.77
b. all - lexicons	60.08*	58.98*	60.51*	59.94*
b.1. all - manual lexicons	66.59*	66.24*	69.52*	67.26*
b.2. all - automatic lexicons	65.17*	64.15*	63.89*	66.46*
b.3. all - Sentiment140 Lexicons	66.84*	66.80*	66.58*	67.61*
b.4. all - Hashtag Sentiment Lexicons	67.65*	67.82	67.64*	71.16*
b.5. all - automatic lexicons of bigrams & non-contiguous pairs	67.65*	66.84	67.44*	69.42
c. all - ngrams	64.07*	65.68*	67.49*	66.93*
c.1. all - word ngrams	66.64*	66.70*	68.29*	67.64*
c.2. all - character ngrams	67.64*	68.28	68.74*	69.11
d. all - POS	67.54*	67.64	70.47	68.42*
e. all - clusters	68.21*	68.33	70.00	68.56*
f. all - encodings (elongated, emoticons, punctuations, all-caps, hashtags)	67.99*	68.66	70.79	69.82

the positive class. We also include the baseline results obtained using an SVM and unigram features alone (row b).

Table 8 shows the results of the ablation experiments where we repeat the same classification process but remove one feature group at a time. The most influential features turn out to be the sentiment lexicon features (row b): they provide gains of 8–10 percentage points on all SemEval-2013 datasets. Note that the contribution of the automatic tweet-specific lexicons (row b.2) substantially exceeds the contribution of the manual lexicons (row b.1). This is especially noticeable on the tweet test set where the use of the automatic lexicons results in improvement of 6.5 percentage points. Also, the use of bigrams and non-contiguous pairs (row b.5) bring additional gains over using only the unigram lexicons.

The second most important feature group for the message-level task is ngrams (row c): word ngrams and character ngrams. Part-of-speech tagging (row d) and clustering (row e) provide only small improvements. Also, removing the sentiment encoding features like hashtags, emoticons, and elongated words (row f) has little impact on performance, but this is probably because the discriminating information in them is also captured by some other features such as character and word ngrams.

Next, we compare the different strategies of processing negation (Table 9). Observe that processing negation benefits the overall sentiment analysis system: all methods we test outperform the baseline of disregarding negation (row a.1). Employing the Affirmative Context Lexicons and the Negated Context Lexicons (row b) provides substantial improvement

Table 9: Message-level task: The macro-averaged F-scores on the Semeval-2013 datasets for different negation processing strategies. Scores marked with * are statistically significantly different ($p < .05$) from the corresponding scores in row c (our best result).

Experiment	**Train. Set**	**Dev. Set**	**Test Sets**	
			Tweets	**SMS**
a. Base automatic lexicons				
a.1. disregarding negation	66.62*	67.36	67.99*	65.29*
a.2. reversing polarity	67.61*	68.04	68.95*	66.96*
b. AffLex and NegLex	68.13*	68.41	69.95*	69.59
c. AffLex and NegLex (Positional)	68.19	68.43	70.45	69.77

Table 10: Message-level task: The results obtained on the movie review excerpts dataset.

System	**Accuracy**
a. Majority baseline	50.1
b. SVM-unigrams	71.9
c. Previous best result (Socher et al., 2013)	85.4
d. Our system	**85.5**

over the standard polarity reversing strategy on the Base Lexicons (row a.2). Replacing the Negated Context Lexicons with the Negated Context (Positional) Lexicons (row c) results in some additional gains for the system.

(b) On the Movie Reviews data: The results obtained using our system on the movie review excerpts dataset is shown in Table 10. Our system, trained on the sentence-level annotations of the training and development subsets, is able to correctly classify 85.5% of the test subset. Note that we ignore the annotations on the word and phrase level as well as the parse tree structure used by Socher et al. (2013). Even on a non-tweet domain, employing the automatically generated, tweet-specific lexicons significantly improves the overall performance: without the use of these lexicons, the performance drops to 83.9%. Furthermore, our system demonstrates the state-of-the-art performance surpassing the previous best result obtained on this dataset (Socher et al., 2013). (The results in rows c and d are not statistically significantly different.)

6.2.2 Results for the Term-level Task

Table 11 shows the performance of our sentiment analysis system on the SemEval-2013 term-level task. Our official submission (row c.1) obtained a macro-averaged F-score of 88.93 on the tweet set and was ranked first among 29 submissions from 23 participating teams.[15] Even with no tuning specific to SMS data, our system ranked second on the SMS test set with an F-score of 88.00. The score of the first ranking system on the SMS set was 88.39. A post-competition bug-fix and the use of the Affirmative Context Lexicons and the

15. The second-best system that used no additional labeled data obtained the score of 86.98 on the tweet test set.

Table 11: Term-level task: The macro-averaged F-scores on the SemEval-2013 datasets.

Classifier	**Train. Set**	**Dev. Set**	**Test Sets**	
			Tweets	**SMS**
a. Majority baseline	38.38	36.34	38.13	32.11
b. SVM-unigrams	78.04	79.76	80.28	78.71
c. Our system:				
c.1. official SemEval-2013 submission	86.80	86.49	88.93	88.00
c.2. best result	87.03	87.07	89.50	88.20

Table 12: Term-level task: The macro-averaged F-scores obtained on the SemEval-2013 datasets when one of the feature groups is removed. Scores marked with * are statistically significantly different ($p < .05$) from the corresponding scores in row a.

Experiment	**Train. Set**	**Dev. Set**	**Test Sets**	
			Tweets	**SMS**
a. all features	87.03	87.07	89.50	88.20
b. all - lexicons	82.77*	81.75*	85.56*	83.52*
b.1. all - manual lexicons	86.16*	86.22	88.21*	87.27*
b.2. all - automatic lexicons	85.28*	85.66*	88.02*	86.39*
c. all - ngrams	84.08*	84.94*	85.73*	82.94*
c.1. all - word ngrams	86.65*	86.30	88.51*	87.02*
c.2. all - char. ngrams	86.67*	87.58	89.20	87.15*
d. all - stopwords	87.07*	87.08	89.42*	88.07*
e. all - encodings (elongated words, emoticons, punctuation, uppercase)	87.11	87.08	89.44	88.17
f. all - target	72.65*	71.72*	74.12*	69.37*
g. all - context	83.76*	83.95*	85.56*	86.63*

Negated Context (Positional) Lexicons resulted in F-score of 89.50 on the tweets set and 88.20 on the SMS set (row c.2). The difference between the best score and the official score on the tweet test set is statistically significant. The table also shows the baseline results obtained by a majority classifier that always predicts the most frequent class as output (row a), and an additional baseline result obtained using an SVM and unigram features alone (row b).

Table 12 presents the results of the ablation experiments where feature groups are alternately removed from the final model. Observe that the sentiment lexicon features (row b) are again the most useful group—removing them leads to a drop in F-score of 4–5 percentage points on all datasets. Both manual (row b.1) and automatic (row b.2) lexicons contribute significantly to the overall sentiment analysis system, with the automatic lexicons consistently showing larger gains.

Table 13: Term-level task: The macro-averaged F-scores obtained on the different subsets of the SemEval-2013 tweet test set with one of the feature groups removed. The number in brackets is the difference with the scores in row a. Scores marked with * are statistically significantly different ($p < .05$) from the corresponding scores in row a.

Classifier	**Targets fully seen in training**	**Targets partially seen in training**	**Targets unseen in training**
a. all features	93.31	85.42	84.09
b. all - lexicons	92.96 (-0.35)	81.26 (-4.16)*	69.55 (-14.54)*
b.1. all - manual lexicons	92.94 (-0.37)	84.51 (-0.91)	79.33 (-4.76)*
b.2. all - automatic lexicons	92.98 (-0.33)	84.08 (-1.34)	79.41 (-4.68)*
c. all - ngrams	89.30 (-4.01)*	81.61 (-3.81)*	80.62 (-3.47)*

The ngram features (row c) are the next most useful group on the term-level task. Note that removing just the word ngram features (row c.1) or just the character ngram features (row c.2) results in only a small drop in performance. This indicates that the two feature groups capture similar information.

The last two rows in Table 12 show the results obtained when the features are extracted only from the context of the target (and not from the target itself) (row f) and when they are extracted only from the target (and not from its context) (row g). Observe that even though the target features are substantially more useful than the context features, adding the context features to the system improves the F-scores by roughly 2 to 4 points.

The performance of the sentiment analysis system is significantly higher in the term-level task than in the message-level task. The difference in performance on these two tasks can also be observed for the SVM-unigrams baseline. We analyzed the provided labeled data to determine why unigrams performed so strongly in the term-level task, and found that most of the test target tokens (85.1%) occur as target tokens in the training data. Further, the distribution of occurrences of a target term in different polarities is skewed towards one polarity or other. On average, a word appears in target phrases of the same polarity 80.8% of the time. These facts explain, at least in part, the high overall result and the dominant role of unigrams in the term-level task. To evaluate the impact of different feature groups on the test data with unseen target terms, we split the SemEval-2013 tweet test set into three subsets. Every instance in the first subset, "targets fully seen in training", has a target X (X can be a single word or a multi-word expression) with the following property: there exist instances in the training data with exactly the same target. The first subset comprises 55% of the test set. Every instance in the second subset, "targets partially seen in training", has a target X with the following property: there exist instances in the training data whose target expression includes one or more, but not all, tokens in X. The second subset comprises 31% of the test set. Every instance in the third subset, "targets unseen in training", has a target X with the following property: there are no instances in the training data whose target includes any of the tokens in X. The third subset comprises 14% of the test set. Table 13 shows the results of the ablation experiments on these three

Table 14: Term-level task: The macro-averaged F-scores on the SemEval-2013 datasets for different negation processing strategies. Scores marked with * are statistically significantly different ($p < .05$) from the corresponding scores in row c (our best result).

Experiment	**Train. Set**	**Dev. Set**	**Test Sets**	
			Tweets	**SMS**
a. Base automatic lexicons				
a.1. disregarding negation	85.88*	86.37*	88.38*	86.77*
a.2. reversing polarity	86.85	86.48*	89.10*	88.34
b. AffLex and NegLex	86.89	86.60*	89.33	87.89
c. AffLex and NegLex (Positional)	87.03	87.07	89.50	88.20

subsets. Observe that on the instances with unseen targets the sentiment lexicons play a more prominent role, providing a substantial gain (14.54 percentage points).

In the next set of experiments, we compare the performance of different approaches to negation handling on the term-level task (Table 14). Similar to the message-level task, processing negation proves beneficial on the term-level task as well. All tested negation processing approaches show better results than the default strategy of disregarding negation (row a.1). The use of the Affirmative Context Lexicons and the Negated Context Lexicons (row b) and especially the Negated Context (Positional) Lexicons (row c) provides additional gains over the results obtained through the use of the polarity reversing method (row a.2).

7. Conclusions

We created a supervised statistical sentiment analysis system that detects the sentiment of short informal textual messages such as tweets and SMS (message-level task) as well as the sentiment of a term (a word or a phrase) within a message (term-level task). The system ranked first in both tasks at the SemEval-2013 competition 'Sentiment Analysis in Twitter'. Moreover, it demonstrated the state-of-the-art performance on two additional datasets: the SemEval-2013 SMS test set and a corpus of movie review excerpts.

In this system, we implemented a variety of features based on surface form and lexical categories. We also included features derived from several sentiment lexicons: (1) existing, manually created, general-purpose lexicons and (2) high-coverage, tweet-specific lexicons that we generated from tweets with sentiment-word hashtags and from tweets with emoticons. Our experiments showed that the new tweet-specific lexicons are superior in sentiment prediction on tweets in both unsupervised and supervised settings.

Processing negation plays an important role in sentiment analysis. Many previous studies adopted a simple technique to reverse polarity of words in the scope of negation. In this work, we demonstrated that this polarity reversing method may not be always appropriate. In particular, we showed that when positive terms are negated, they tend to convey a negative sentiment. In contrast, when negative terms are negated, they tend to still convey a negative sentiment. Furthermore, the evaluative intensity for both positive and negative terms changes in a negated context, and the amount of change varies from

term to term. To adequately capture the impact of negation on individual terms, we proposed to empirically estimate the sentiment scores of terms in negated context from large tweet corpora, and built two lexicons, one for terms in negated contexts and one for terms in affirmative (non-negated) contexts. By using these Affirmative Context Lexicons and Negated Context Lexicons we were able to significantly improve the performance of the overall sentiment analysis system on both tasks. In particular, the features derived from these lexicons provided gains of up to 6.5 percentage points over the other feature groups.

Our system can process 100 tweets in a second. Thus, it is suitable for small- and big-data versions of applications listed in the introduction. We recently annotated 135 million tweets over a cluster of 50 machines in 11 hours. We have already employed the sentiment analysis system within larger systems for detecting intentions behind political tweets (Mohammad, Kiritchenko, & Martin, 2013), and for detecting emotions in text (Mohammad & Kiritchenko, 2014). We are also interested in applying and evaluating the lexicons generated from tweets on data from other kinds of text such as blogs and news articles. We are developing a way to identify sentiment towards particular aspects of target entities. We also plan to adapt our sentiment analysis system to languages other than English. Along the way, we continue to improve the sentiment lexicons by generating them from larger amounts of data, and from different kinds of data, such as tweets, blogs, and Facebook posts. We are especially interested in algorithms that gracefully handle all kinds of sentiment modifiers such as negations, intensifiers (e.g., *very*, *hardly*), and discourse connectives (e.g., *but*, *however*).

Acknowledgments

We thank Colin Cherry for providing his SVM code and for helpful discussions.

References

Agarwal, A., Xie, B., Vovsha, I., Rambow, O., & Passonneau, R. (2011). Sentiment analysis of twitter data. In *Proceedings of the Workshop on Languages in Social Media*, LSM '11, pp. 30–38, Portland, Oregon.

Aisopos, F., Papadakis, G., Tserpes, K., & Varvarigou, T. (2012). Textual and contextual patterns for sentiment analysis over microblogs. In *Proceedings of the 21st International Conference on World Wide Web Companion*, WWW '12 Companion, pp. 453–454, New York, NY, USA.

Baccianella, S., Esuli, A., & Sebastiani, F. (2010). SentiWordNet 3.0: an enhanced lexical resource for sentiment analysis and opinion mining. In *Proceeding of the 7th International Conference on Language Resources and Evaluation*, Vol. 10 of *LREC '10*, pp. 2200–2204.

Bakliwal, A., Arora, P., Madhappan, S., Kapre, N., Singh, M., & Varma, V. (2012). Mining sentiments from tweets. In *Proceedings of the 3rd Workshop on Computational Approaches to Subjectivity and Sentiment Analysis*, WASSA '12, pp. 11–18, Jeju, Republic of Korea.

Bellegarda, J. (2010). Emotion analysis using latent affective folding and embedding. In *Proceedings of the NAACL-HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, Los Angeles, California.

Boucher, J. D., & Osgood, C. E. (1969). The Pollyanna Hypothesis. *Journal of Verbal Learning and Verbal Behaviour*, *8*, 1–8.

Boucouvalas, A. C. (2002). Real time text-to-emotion engine for expressive internet communication. *Emerging Communication: Studies on New Technologies and Practices in Communication*, *5*, 305–318.

Bougie, J. R. G., Pieters, R., & Zeelenberg, M. (2003). Angry customers don't come back, they get back: The experience and behavioral implications of anger and dissatisfaction in services. Open access publications, Tilburg University.

Brody, S., & Diakopoulos, N. (2011). Cooooooooooooooollllllllllllll!!!!!!!!!!!!!!: using word lengthening to detect sentiment in microblogs. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, pp. 562–570, Stroudsburg, PA, USA. Association for Computational Linguistics.

Chang, C.-C., & Lin, C.-J. (2011). LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, *2*(3), 27:1–27:27.

Chew, C., & Eysenbach, G. (2010). Pandemics in the age of Twitter: Content analysis of tweets during the 2009 H1N1 outbreak. *PLoS ONE*, *5*(11), e14118+.

Choi, Y., & Cardie, C. (2008). Learning with compositional semantics as structural inference for subsentential sentiment analysis. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '08, pp. 793–801.

Davidov, D., Tsur, O., & Rappoport, A. (2010). Enhanced sentiment learning using Twitter hashtags and smileys. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, COLING '10, pp. 241–249, Beijing, China.

Esuli, A., & Sebastiani, F. (2006). SENTIWORDNET: A publicly available lexical resource for opinion mining. In *In Proceedings of the 5th Conference on Language Resources and Evaluation*, LREC '06, pp. 417–422.

Francisco, V., & Gervás, P. (2006). Automated mark up of affective information in English texts. In Sojka, P., Kopecek, I., & Pala, K. (Eds.), *Text, Speech and Dialogue*, Vol. 4188 of *Lecture Notes in Computer Science*, pp. 375–382. Springer Berlin / Heidelberg.

Genereux, M., & Evans, R. P. (2006). Distinguishing affective states in weblogs. In *Proceedings of the AAAI Spring Symposium on Computational Approaches to Analysing Weblogs*, pp. 27–29, Stanford, California.

Gimpel, K., Schneider, N., O'Connor, B., Das, D., Mills, D., Eisenstein, J., Heilman, M., Yogatama, D., Flanigan, J., & Smith, N. A. (2011). Part-of-speech tagging for Twitter: Annotation, features, and experiments. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, ACL '11.

Go, A., Bhayani, R., & Huang, L. (2009). Twitter sentiment classification using distant supervision. Tech. rep., Stanford University.

Hanley, J., & McNeil, B. (1982). The meaning and use of the area under a Receiver Operating Characteristic (ROC) curve. *Radiology*, *143*, 29–36.

Hatzivassiloglou, V., & McKeown, K. R. (1997). Predicting the semantic orientation of adjectives. In *Proceedings of the 8th Conference of European Chapter of the Association for Computational Linguistics*, EACL '97, pp. 174–181, Madrid, Spain.

Hu, M., & Liu, B. (2004). Mining and summarizing customer reviews. In *Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '04, pp. 168–177, New York, NY, USA. ACM.

Jansen, B. J., Zhang, M., Sobel, K., & Chowdury, A. (2009). Twitter power: Tweets as electronic word of mouth. *Journal of the American Society for Information Science and Technology*, *60*(11), 2169–2188.

Jia, L., Yu, C., & Meng, W. (2009). The effect of negation on sentiment analysis and retrieval effectiveness. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, CIKM '09, pp. 1827–1830, New York, NY, USA. ACM.

Jiang, L., Yu, M., Zhou, M., Liu, X., & Zhao, T. (2011). Target-dependent Twitter sentiment classification. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, ACL '11, pp. 151–160.

John, D., Boucouvalas, A. C., & Xu, Z. (2006). Representing emotional momentum within expressive internet communication. In *Proceedings of the 24th International Conference on Internet and Multimedia Systems and Applications*, pp. 183–188, Anaheim, CA. ACTA Press.

Kennedy, A., & Inkpen, D. (2005). Sentiment classification of movie and product reviews using contextual valence shifters. In *Proceedings of the Workshop on the Analysis of Informal and Formal Information Exchange during Negotiations*, Ottawa, Ontario, Canada.

Kennedy, A., & Inkpen, D. (2006). Sentiment classification of movie reviews using contextual valence shifters. *Computational Intelligence*, *22*(2), 110–125.

Kouloumpis, E., Wilson, T., & Moore, J. (2011). Twitter sentiment analysis: The Good the Bad and the OMG!. In *Proceedings of the 5th International AAAI Conference on Weblogs and Social Media*.

Lapponi, E., Read, J., & Ovrelid, L. (2012). Representing and resolving negation for sentiment analysis. In Vreeken, J., Ling, C., Zaki, M. J., Siebes, A., Yu, J. X., Goethals, B., Webb, G. I., & Wu, X. (Eds.), *ICDM Workshops*, pp. 687–692. IEEE Computer Society.

Li, J., Zhou, G., Wang, H., & Zhu, Q. (2010). Learning the scope of negation via shallow semantic parsing. In *Proceedings of the 23rd International Conference on Computational Linguistics*, COLING '10, pp. 671–679, Beijing, China.

Liu, B., & Zhang, L. (2012). A survey of opinion mining and sentiment analysis. In Aggarwal, C. C., & Zhai, C. (Eds.), *Mining Text Data*, pp. 415–463. Springer US.

Liu, H., Lieberman, H., & Selker, T. (2003). A model of textual affect sensing using real-world knowledge. In *Proceedings of the 8th International Conference on Intelligent User Interfaces*, IUI '03, pp. 125–132, New York, NY. ACM.

Mandel, B., Culotta, A., Boulahanis, J., Stark, D., Lewis, B., & Rodrigue, J. (2012). A demographic analysis of online sentiment during hurricane Irene. In *Proceedings of the Second Workshop on Language in Social Media*, LSM '12, pp. 27–36, Stroudsburg, PA, USA. Association for Computational Linguistics.

Mihalcea, R., & Liu, H. (2006). A corpus-based approach to finding happiness. In *Proceedings of the AAAI Spring Symposium on Computational Approaches to Analysing Weblogs*, pp. 139–144. AAAI Press.

Mohammad, S., Kiritchenko, S., & Zhu, X. (2013). NRC-Canada: Building the state-of-the-art in sentiment analysis of tweets. In *Proceedings of the International Workshop on Semantic Evaluation*, SemEval '13, Atlanta, Georgia, USA.

Mohammad, S. (2012). #emotional tweets. In *Proceedings of the First Joint Conference on Lexical and Computational Semantics*, *SEM '12, pp. 246–255, Montréal, Canada. Association for Computational Linguistics.

Mohammad, S., Dunne, C., & Dorr, B. (2009). Generating high-coverage semantic orientation lexicons from overtly marked words and a thesaurus. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing: Volume 2*, EMNLP '09, pp. 599–608.

Mohammad, S. M., & Kiritchenko, S. (2014). Using hashtags to capture fine emotion categories from tweets. *To appear in Computational Intelligence*.

Mohammad, S. M., Kiritchenko, S., & Martin, J. (2013). Identifying purpose behind electoral tweets. In *Proceedings of the 2nd International Workshop on Issues of Sentiment Discovery and Opinion Mining*, WISDOM '13, pp. 1–9.

Mohammad, S. M., & Turney, P. D. (2010). Emotions evoked by common words and phrases: Using Mechanical Turk to create an emotion lexicon. In *Proceedings of the NAACL-HLT Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, LA, California.

Mohammad, S. M., & Yang, T. W. (2011). Tracking sentiment in mail: How genders differ on emotional axes. In *Proceedings of the ACL Workshop on Computational Approaches to Subjectivity and Sentiment Analysis*, WASSA '11, Portland, OR, USA.

Neviarouskaya, A., Prendinger, H., & Ishizuka, M. (2011). Affect analysis model: novel rule-based approach to affect sensing from text. *Natural Language Engineering*, *17*, 95–135.

Pak, A., & Paroubek, P. (2010). Twitter as a corpus for sentiment analysis and opinion mining. In *Proceedings of the 7th Conference on International Language Resources and Evaluation*, LREC '10, Valletta, Malta. European Language Resources Association (ELRA).

Pang, B., & Lee, L. (2005). Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, ACL '05, pp. 115–124.

Pang, B., & Lee, L. (2008). Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*, *2*(1–2), 1–135.

Pang, B., Lee, L., & Vaithyanathan, S. (2002). Thumbs up?: Sentiment classification using machine learning techniques. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '02, pp. 79–86, Philadelphia, PA.

Polanyi, L., & Zaenen, A. (2004). Contextual valence shifters. In *Exploring Attitude and Affect in Text: Theories and Applications (AAAI Spring Symposium Series)*.

Porter, M. (1980). An algorithm for suffix stripping. *Program*, *3*, 130–137.

Ravaja, N., Saari, T., Turpeinen, M., Laarni, J., Salminen, M., & Kivikangas, M. (2006). Spatial presence and emotions during video game playing: Does it matter with whom you play?. *Presence: Teleoperators and Virtual Environments*, *15*(4), 381–392.

Salathé, M., & Khandelwal, S. (2011). Assessing vaccination sentiments with online social media: Implications for infectious disease dynamics and control. *PLoS Computational Biology*, *7*(10).

Sauper, C., & Barzilay, R. (2013). Automatic aggregation by joint modeling of aspects and values. *Journal of Artificial Intelligence Research*, *46*, 89–127.

Schwartz, H., Eichstaedt, J., Kern, M., Dziurzynski, L., Lucas, R., Agrawal, M., Park, G., Lakshmikanth, S., Jha, S., Seligman, M., & Ungar, L. (2013). Characterizing geographic variation in well-being using tweets. In *Proceedings of the International AAAI Conference on Weblogs and Social Media*.

Socher, R., Huval, B., Manning, C. D., & Ng, A. Y. (2012). Semantic compositionality through recursive matrix-vector spaces. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '12. Association for Computational Linguistics.

Socher, R., Perelygin, A., Wu, J. Y., Chuang, J., Manning, C. D., Ng, A. Y., & Potts, C. (2013). Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '13. Association for Computational Linguistics.

Stone, P., Dunphy, D. C., Smith, M. S., Ogilvie, D. M., & associates (1966). *The General Inquirer: A Computer Approach to Content Analysis*. The MIT Press.

Taboada, M., Brooke, J., Tofiloski, M., Voll, K., & Stede, M. (2011). Lexicon-based methods for sentiment analysis. *Computational Linguistics*, *37*(2), 267–307.

Thelwall, M., Buckley, K., & Paltoglou, G. (2011). Sentiment in Twitter events. *Journal of the American Society for Information Science and Technology*, *62*(2), 406–418.

Turney, P., & Littman, M. L. (2003). Measuring praise and criticism: Inference of semantic orientation from association. *ACM Transactions on Information Systems*, *21*(4).

Velásquez, J. D. (1997). Modeling emotions and other motivations in synthetic agents. In *Proceedings of the 14th National Conference on Artificial Intelligence and 9th Conference on Innovative Applications of Artificial Intelligence*, AAAI'97/IAAI'97, pp. 10–15. AAAI Press.

Verma, S., Vieweg, S., Corvey, W., Palen, L., Martin, J., Palmer, M., Schram, A., & Anderson, K. (2011). Natural language processing to the rescue? Extracting "situational

awareness" tweets during mass emergency. In *Proceedings of the International AAAI Conference on Weblogs and Social Media*.

Wiegand, M., Balahur, A., Roth, B., Klakow, D., & Montoyo, A. (2010). A survey on the role of negation in sentiment analysis. In *Proceedings of the Workshop on Negation and Speculation in Natural Language Processing*, NeSp-NLP '10, pp. 60–68, Stroudsburg, PA, USA. Association for Computational Linguistics.

Wilson, T., Kozareva, Z., Nakov, P., Rosenthal, S., Stoyanov, V., & Ritter, A. (2013). SemEval-2013 Task 2: Sentiment analysis in Twitter. In *Proceedings of the International Workshop on Semantic Evaluation*, SemEval '13, Atlanta, Georgia, USA.

Wilson, T., Wiebe, J., & Hoffmann, P. (2005). Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, HLT '05, pp. 347–354, Stroudsburg, PA, USA. Association for Computational Linguistics.

Yeh, A. (2000). More accurate tests for the statistical significance of result differences. In *Proceedings of the 18th conference on Computational linguistics - Volume 2*, COLING '00, pp. 947–953, Stroudsburg, PA, USA. Association for Computational Linguistics.