# Negation Scope Detection for Twitter Sentiment Analysis

**Johan Reitan** **Jørgen Faret** **Björn Gambäck** **Lars Bungum**
{johan.reitan,jorgenfar}@gmail.com {gamback,larsbun}@idi.ntnu.no
Department of Computer and Information Science
Norwegian University of Science and Technology
Sem Sælands vei 7–9, NO–7491 Trondheim, Norway

## Abstract

The paper describes the first sophisticated negation scope detection system for Twitter sentiment analysis. The system has been evaluated both on existing corpora from other domains and on a corpus of English Twitter data (*tweets*) annotated for negation. It produces better results than what has been reported in other domains and improves the performance on tweets containing negation when incorporated into a state-of-the-art Twitter sentiment analyser.

## 1 Introduction

Exploring public opinion on various subjects has always been an important part of humans' information gathering behaviour. Where one in the past needed to conduct surveys to learn about opinion trends, the availability of online data expressing sentiment has allowed for non-intrusive data mining to extract this information. Over the last decade, there has been a substantial increase in the amount of work done in the field of sentiment analysis, which has largely followed the available data, in recent years shifting the field towards Twitter data, i.e., towards Twitter sentiment analysis.

Twitter[1] is a micro-blogging site that allows users to write textual entries ('*tweets*') of up to 140 characters. The tweets are available through Twitter's API and represent a real-time information stream of opinionated data. Performing language processing on tweets presents new challenges because of their informal nature. Tweets often contain misspellings, slang and abbreviations, and unconventional linguistic means, such as capitalization or elongation of words to show emphasis. Additionally, tweets contain special features like *emoticons* and *hashtags* that may have analytical value.

[1] https://www.twitter.com

The ability to handle linguistic negation of terms is an important aspect of sentiment classification. The *valence* of a segment of text (its communicated positive or negative attitude) can be equated to its sentimental orientation, and valence shifters are terms that change the sentimental orientation of other terms. In sentiment analysis, negators often act as valence shifters, since flipping a proposition's truth value significantly shifts, or reverses, the valence it conveys. Givón (1993) defines two forms of grammatical negation: *morphological*, where individual words are negated with an affix, and *syntactic*, where a set of words is negated by a word or phrase; the topic of the present paper. Negators in syntactical negation, known as *negation cues* or *negation signals*, function as operators, with an associated affected scope of words (Morante and Sporleder, 2012). The most common negation cue in English is *not*, along with its contractions, such as *couldn't* or *isn't* (Tottie, 1991).

Negation classifiers have been developed for other domains with dramatic performance improvements (Section 2). However, almost all state-of-the-art Twitter sentiment analysis systems use a simple approach of marking as negated all terms from a negation cue to the next punctuation (Section 3). We present this simple model as a baseline, but improve on it by introducing sophisticated negation scope detection for Twitter sentiment analysis.

Several negation-annotated corpora are available, but none for the Twitter domain. To be able to evaluate Twitter negation detection and to train supervised machine learning classifiers, a corpus of tweets was annotated with negation meta-data. The new and existing corpora are detailed in Section 4.

Sections 5 and 6 describe the construction of two classification systems: a Twitter negation scope detector and a state-of-the-art sentiment analyser incorporating the negation classifier, as well as experiments conducted on the two systems. Section 7 summarizes the results and suggests future work.



*Proceedings of the 6th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis (WASSA 2015)*, pages 99–108,
Lisboa, Portugal, 17 September, 2015. 

## 2 Negation Scope Detection, NSD

The main application area of identifying the scope of negation, or negation scope detection (NSD), was originally biomedical texts, such as clinical reports and discharge summaries, but has in recent times shifted towards sentiment analysis (SA). Early solutions were typically rule-based, such as the NegFinder (Mutalik et al., 2001) and NegEx (Chapman et al., 2001) systems, that both heavily incorporated the use of regular expressions. NSD was the focus of a shared task at *SEM 2012 (Morante and Blanco, 2012), and in 2010 CoNLL included a similar sub-task on detecting speculation cues and their affected scope (Farkas et al., 2010). Most well-performing submissions to both tasks used supervised machine learning approaches.

Morante and Daelemans (2009) developed an NSD system that uses meta-learning for classification. They applied this approach to the CoNLL'10 shared task and achieved the best $F_1$-score of all participating teams. The tokens were first tagged and split into chunks, and the main algorithm then consisted of two steps: signal identification (negation cue detection) and scope identification. For the first phase, Morante and Daelemans (2009) used a decision tree to classify if a token is at the beginning, inside or outside a negation signal. In the second phase, a Conditional Random Fields (CRF)-based meta-learner predicted scope classes based on the output from three classifiers, a Support Vector Machine (SVM), a k-nearest neighbour classifier and a CRF classifier. Zhu et al. (2010) also worked on biomedical texts and proposed a rule-based shallow semantic parsing solution: they set the negation signal as the predicate, and then use an SVM-based binary classifier to find the negated scope by identifying the correct argument(s).

Wiegand et al. (2010) surveyed the effects of NSD on SA, concluding it to be "highly relevant". Moilanen and Pulman (2007) built an SA system with a sophisticated NSD mechanism focused on syntactic composition. Potts (2011) achieved $\sim$12 % better accuracy with a simple model marking as negated all terms from a detected negation cue to the next punctuation. Councill et al. (2010) utilized the MaltParser (Nivre et al., 2007) for tokenization, part-of-speech-tagging, and creation of a dependency tree to generate a feature vector for training a CRF classifier. Tested on consumer reviews, the classifier improved $F_1$ scores by 29.5 % and 11.4 % on positive resp. negative sentiments.

## 3 Twitter Sentiment Analysis, TSA

The typical approach to Twitter sentiment analysis (TSA) is a supervised machine learning system with three main steps: preprocessing, feature extraction, and classification. Preprocessing aims to reduce noise and consists of a variety of filters, e.g., to normalize user mentions by substituting them with the tag `||T||` and URLs with `||U||` (Agarwal et al., 2011), prefix *retweets* (reposts of previous content) with "RT", and substitute letters that occur many times sequentially in elongated words (e.g., *happyyyyyy*) with one or two occurrences of the letter. It was previously common to filter out hashtags (Selmer et al., 2013, e.g), since those when used as intended (to categorize posts by topic) offer little sentiment information; however, Mohammad (2012) shows that they add sentiment by indicating the tone of the message or the writer's emotions.

Much recent progress in the field has been in connection to the International Workshop on Semantic Evaluation (SemEval), which since 2013 (Nakov et al., 2013) has included shared tasks on Sentiment Analysis in Twitter, with expression-level subtasks, to correctly classify the overall polarity of whole tweets. Many later systems have based their feature matrix on the SemEval'13 top submission (Mohammad et al., 2013). Tang et al. (2014) define it as the state-of-the-art feature set ('STATE'). This set will be further discussed in Section 6.1, but includes most typically used features such as word and character $n$-grams, different types of token frequencies, and a set of prior polarity sentiment lexica.

Most well-performing systems for TSA use a supervised machine learning-based classifier. An analysis of the classification algorithms used by the ten top ranking submissions to SemEval'14 (Rosenthal et al., 2014) shows that SVM and Logistic Regression were the most popular choices.

Few state-of-the-art TSA systems address negation systematically, but rather use the simple model described by Potts (2011), to assign a negation cue scope over all terms to the next punctuation. So do the top-3 SemEval'14 systems (Miura et al., 2014; Tang et al., 2014; Günther et al., 2014) and almost all SemEval'15 systems treating negation, including two of the top-3 (Hagen et al., 2015; Hamdan et al., 2015), although Rosenthal et al. (2015) mention negation as one area the systems focused on.

If the model includes prior polarity lexica, just inverting the sentiment polarity of negated terms is incorrect (Kiritchenko et al., 2014): positive terms

when negated tend to shift polarity and decrease their intensity, while negative terms mostly stay negative with reduced intensity. Kiritchenko et al. (2014) thus created tweet-specific sentiment lexica containing scores for terms in affirmative and negated contexts: *NRC Hashtag Sentiment Lexicon* and *Sentiment140 Lexicon*. The lexica added clear performance improvements (5.83 % average $F_1$ increase over the five SemEval'14 data sets), even though the negated contexts were simply assumed to be from a negation cue to the next punctuation.

Plotnikova et al. (2015) created one of the better SemEval'15 systems by using the heuristic of assigning a negation cue scope over the 4 next tokens, which compares well with the 3.8 average tokens in the negation scope for our Twitter Negation Corpus (Table 1). Only one SemEval'15 system utilized an alternative treatment: Cerezo-Costas and Celix-Salgado (2015) trained a CRF-based classifier to detect the scope of what they call "denier particles" (i.e., negation) and "reversal verbs" (e.g., 'avoid', 'prevent'), that reverse the polarity of the terms in their scope. The system did not perform well over all, but ranked #1 on the 2014 tweet sarcasm data.

## 4 Data Sets

There are negation scope corpora available for other domains and sentiment-annotated data available from the SemEval TSA tasks. However, performing NSD using supervised machine learning requires a set of tweets annotated for negation cues and scopes, so such a corpus was also developed. The new and existing data sets are described below.

**BioScope Corpus** is a collection of bio-medical texts annotated for speculation and negation (Vincze et al., 2008). It consists of three subcorpora: medical free texts (6,383 sentences), biological full papers (2,670), and biological scientific abstracts (11,871). The free text part differs significantly from the others in that it contains mainly short and concise sentences. The rate of negation, though, is even across the entire corpus: 13.6 % of the sentences in the free texts, 12.7 % in the full papers, and 13.5 % in the abstracts contain negation.

**SFU Review Corpus** contains 400 reviews (50 each from 8 domains such as movies and consumer products) annotated at the token level for negation and speculation by Simon Fraser University (Konstantinova et al., 2012). In total, it consists of 17,263 sentences, with 18.1 % containing negation.

| | |
|---|---|
| Number of tweets | 4,000 |
| Total number of tokens | 61,172 |
| Average tokens per tweet | 15.3 |
| Average tokens per sentence | 10.2 |
| Tweets containing negation | 539 |
| Total number of scopes | 615 |
| Average cues per negated tweet | 1.14 |
| Average tokens in scope | 3.8 |

Table 1: Twitter Negation Corpus

**SemEval Twitter sentiment analysis data** have been annotated using Mechanical Turk, and include training, development and test sets, as well as out-of-domain test sets. Due to Twitter's privacy policy, the data cannot be distributed directly, but is downloaded with a script that uses tweet IDs to match tweets with their sentiment labels. Tweets that have been deleted since the data sets' creation are unavailable, and the sets grow smaller over time. The total size of the SemEval'14 data when downloaded by us, in November 2014, was 12,754 tweets.

**Twitter Negation Corpus** contains 4,000 tweets downloaded through Twitter's API and annotated by two of the authors using a web application developed for this purpose. The application retrieves a tokenized tweet from the database and displays it as a container of HTML buttons, where each button represents a token. The user clicks a token to mark it as a negation cue and on corresponding tokens to mark the scope. Inter-annotator agreement was calculated at token and full scope level. The token level score is the number of tokens annotators agree on divided by the total number of tokens. It is an unbalanced measure as tokens in affirmative contexts greatly outnumber those in negated. Full scope agreement entails that annotator scopes match completely. Token level agreement was 98.9 % and full scope agreement 73.8 %. All scope conflicts were reviewed and resolved after discussion.

Statistics for the corpus are shown in Table 1, with figures relating to negation in the lower half. Tottie (1991) states that the frequency of negation in written English is 12.8 %, and the fraction of tweets containing negation, 13.5 % (539/4000) is quite close to that. The average number of tokens per sentence is 10.2 and the average scope size is 3.8. For comparison, the equivalent numbers of the full paper BioScope sub-corpus are 26.2 and 8.8 (Morante and Daelemans, 2009), which indicates that simpler language is used in the Twitter corpus.

| | | | | |
|---|---|---|---|---|
| aint | *cannot* | cant | *darent* | didnt |
| doesnt | *dont* | hadnt | *hardly* | hasnt |
| havent | *havnt* | isnt | *lack* | *lacking* |
| *lacks* | neither | never | no | *nobody* |
| none | *nor* | not | nothing | nowhere |
| *mightnt* | *mustnt* | *neednt* | *oughtnt* | *shant* |
| shouldnt | *wasnt* | *without* | wouldnt | *n't |

Table 2: Lexicon of negation cues

## 5 Negation Scope Detection Experiments

Two classifiers were created: one to detect the scope of negation and one to assign sentiment. The negation classifier was used in the feature extraction process for the sentiment classifier (Section 6).

Negation scope detection (NSD) is a binary classification problem, where each token is determined to be either in an affirmative or a negated context. For NSD experiments, we report precision (P), recall (R), $F_1$ score, and the *percentage of correctly classified scopes* (PCS): For classification tasks where the output is a sequence, metrics that only consider individual units regardless of their order are often insufficient. PCS measures the accuracy of a scope classifier: a scope is considered correctly classified if, for a given negation cue, every token in its associated scope has been correctly marked.

### 5.1 Negation Classifier Architecture

The classification algorithm consists of two steps: negation cue detection and scope identification. Cue detection is performed by a pattern-matching approach with a lexicon of explicit cues adopted from Councill et al. (2010), as shown in Table 2, where **n't* matches all strings with the suffix *n't*. Note that this list is more extensive than the one of Potts (2011), used in many SemEval systems. Four cues on Potts' list are not in Table 2 (*noone, couldnt, wont, arent*), while the 17 cues in italics are not listed by Potts. An inspection of the 37 cues appearing in the Twitter Negation Corpus revealed seven more cues / spelling variants included on neither list (*idk, dnt, cudnt, ain, eint, neva, neeeever*).

Tweets are preprocessed with the TweeboParser dependency parser (Kong et al., 2014), that performs tokenisation, part-of-speech tagging and parsing, labeling each token with its dependency head. A dependency-based binary CRF classifier then for each token determines whether it is in a negated or affirmative context. The CRF implementation by Okazaki (2007) is used, with a Python binding created by Peng and Korobov (2014).

| Feature | Description |
|---|---|
| Word | lower-case token string |
| POS | part-of-speech tag of the token |
| DRight | distance to nearest negation cue to the right |
| DLeft | distance to nearest negation cue to the left |
| DepD | number of edges to nearest negation cue |
| Dep1POS | POS tag of the 1st order dependency head |
| Dep1D | number of edges to nearest negation cue from the 1st order dependency head |
| Dep2POS | POS tag of the 2nd order dependency head |
| Dep2D | number of edges to nearest negation cue from the 2nd order dependency head |

Table 3: Negation classifier feature set

The classifier is a Twitter-tailored version of the system described by Councill et al. (2010) with one change: the dependency distance from each token to the closest negation cue has been added to the feature set, which is shown in Table 3. The distances (DRight and DLeft) are the minimun linear token-wise distances, i.e., the number of tokens from one token to another. Dependency distance (DepD) is calculated as the minimum number of edges that must be traversed in a dependency tree to move from one token, to another. The classifier takes a parameter, *max distance*, that specifies the maximum distance to be considered (all longer distances are treated as being equivalent). This applies to both linear distance and dependency distance.

### 5.2 Negation Cue Detection

The created Conditional Random Fields negation classifier was evaluated on the Twitter Negation Corpus. The data set was split into two subsets: a development set and an evaluation set. The development set consists of 3,000 tweets and the evaluation set of 1,000 tweets. To ensure more reliable training and testing, given the heavy label imbalance of the corpus, the split was stratified, with the same ratio of tweets containing negation in both subsets.

The actual negation cues in the annotated training data are used when training the classifier, but a lexicon-based cue detection approach is taken during classification. When applied to the Twitter Negation Corpus, the cue detector achieved a precision of 0.873 with a recall of 0.976, and hence an $F_1$ score of 0.922. In comparison, Morante and Daelemans (2009) use a list of negation cues extracted from their training data and thus have perfect cue detection precision, but recall varying from 0.957 (full papers) to 0.987 (abstracts) on the three BioScope sub-corpora.

| Data | NSD model | P | R | $F_1$ | PCS |
|---|---|---|---|---|---|
| Test | Sophisticated | 0.972 | 0.923 | 0.853 | 64.5 |
| | Gold standard | 0.841 | 0.956 | 0.895 | 66.3 |
| | Simple | 0.591 | 0.962 | 0.733 | 43.1 |
| Train | Sophisticated | 0.849 | 0.891 | 0.868 | 66.3 |

Table 4: Negation classifier performance

Inspection of the cue detection output reveals that the classifier mainly struggles with the separation of words used both as negators and exclamations. By far the most significant of these is *no*, with 35 of its 90 occurrences in the corpus being as a non-cue; often it occurs as a determiner functioning as a negator (e.g., "there were no letters this morning"), but it may occur as an exclamation (e.g., "No, I'm not ready yet" and "No! Don't touch it").

Despite the high recall, cue outliers such as *dnt neva*, or *cudnt* could potentially be detected by using word-clusters. We expanded the lexicon of negation cues to contain the whole set of Tweet NLP word clusters created by Owoputi et al. (2013) for each lexical item. Recall was slightly increased, to 0.992, but precision suffered a dramatic decrease to 0.535, since the clusters are too inclusive. More finely-grained word clusters could possibly increase recall without hurting precision.

### 5.3 NSD Classifier Performance

To determine the optimal parameter values, a 7-fold stratified cross validation grid search was performed on the development set over the L1 and L2 CRF penalty coefficients, $C1$ and $C2$ with a parameter space of $\{10^{-4}, 10^{-3}, 10^{-2}, 0.1, 1, 10\}$, in addition to *max distance* (see Section 5.1) with a $[5, 10]$ parameter space. The identified optimal setting was $C1 = 0.1, C2 = 1$, *max distance* $= 7$.

The performance of the *sophisticated* negation scope classifier with the parameter set selected through grid search was evaluated on the held-out test data. The classifier was also tested on the same evaluation set with *gold standard* cue detection (i.e., with perfect negation signal identification).

To establish a baseline for negation scope detection on the Twitter Negation Corpus, we also implemented the simple model described in Section 2 and used by almost all SemEval TSA systems handling negation: When a negation cue is detected, all terms from the cue to the next punctuation are considered negated. Note though, that by using an extended cue dictionary, our *simple* baseline potentially slightly improves on state-of-the-art models.

| Data | Classifier | P | R | $F_1$ | PCS |
|---|---|---|---|---|---|
| SFU | Sophisticated | 0.668 | 0.874 | 0.757 | 43.5 |
| BioScope full | CRF | 0.808 | 0.708 | 0.755 | 53.7 |
| | MetaLearn | 0.722 | 0.697 | 0.709 | 41.0 |
| | Sophisticated | 0.660 | 0.610 | 0.634 | 42.6 |
| | Simple | 0.583 | 0.688 | 0.631 | 43.7 |
| | SSP | 0.582 | 0.563 | 0.572 | 64.0 |

Table 5: Out-of-domain NSD performance

Results from the test run on the evaluation data, and the test on the evaluation set with gold standard cue detection are shown in Table 4, together with the simple baseline, as well as a 7-fold cross validation on the development set.

The classifier achieves very good results. The run on the evaluation set produces an $F_1$ score of 0.853, which is considerably higher than the baseline. It also outperforms Councill et al. (2010) who achieved an $F_1$ score of 0.800 when applying a similar system to their customer review corpus.

### 5.4 Out-of-Domain Performance

Although the negation classifier is a Twitter-tailored implementation of the system described by Councill et al. (2010) with minor modifications the use of a different CRF implementation, POS-tagger and dependency parser may lead to considerable performance differences. To explore the out-of-domain capacity of the classifier, it was evaluated on the SFU Review corpus and the biological full paper part of BioScope, as that sub-corpus has proved to be difficult for negation identification.

Table 5 shows the 5-fold cross-validated performance of the sophisticated negation scope identifier on both corpora, as well as the simple baseline on Bioscope together with the results reported on the same data for the approaches described in Section 2. 'CRF' denotes the CRF-based system from Councill et al. (2010), 'MetaLearn' the meta-learner of Morante and Daelemans (2009), and 'SSP' the shallow semantic parsing solution by Zhu et al. (2010).

As can be seen, the twitter-trained sophisticated negation classifier performs reasonably well on the SFU Review Corpus, but struggles when applied to BioScope, as expected. It is outperformed in terms of $F_1$ score by Councill et al. (2010) and Morante and Daelemans (2009), but reaches a slightly better PCS than the latter system. The modest $F_1$ score is likely caused by the use of upstream preprocessing tools tailored towards Twitter language, which differs significantly from that of biomedical texts.

Notably, the simple model is a strong baseline, which actually outperforms the shallow parser on $F_1$ score and the meta-learner on percentage of correctly classified scopes (PCS).

## 6 An NSD-enhanced Sentiment Classifier

The Twitter sentiment analysis includes three steps: preprocessing, feature extraction, and either training the classifier or classifying samples. A Support Vector Machine classifier is used as it is a state-of-the-art learning algorithm proven effective on text categorization tasks, and robust on large feature spaces. We employ the SVM implementation `SVC` from `Scikit-learn` (Pedregosa et al., 2011), which is based on libsvm (Chang and Lin, 2011).

### 6.1 Sentiment Classifier Architecture

The preprocessing step substitutes newline and tab characters with spaces, user mentions with the string "*@someuser*", and URLs with "http://someurl" using a slightly modified regular expression by *@stephenhay*,[2] matching URLs starting with protocol specifiers or only "www".

The feature extraction step elicitates characteristics based on the STATE set, as shown in Table 6; the top four features are affected by linguistic negation, the rest are not. There are two term frequency-inverse document frequency (TF-IDF) vectorizers, for *word* $n$*-grams* ($1 \leq n \leq 4$) and for *character* $n$*-grams* ($3 \leq n \leq 5$). Both ignore common English stop words, convert all characters to lower case, and select the 1,000 features with highest TF-IDF scores. Tokens in a negation scope are appended the string `_NEG`. The *negated tokens* feature is simply a count of the tokens in a negated context.

The *NRC Hashtag Sentiment Lexicon* and *Sentiment140 Lexicon* (Kiritchenko et al., 2014) contain sentiment scores for words in negated contexts. For lookups, the first negated word in a negation scope is appended with `_NEGFIRST`, and the rest with `_NEG`. The sentiment lexica feature vectors are adopted from Kiritchenko et al. (2014) and contain the number of tokens with $score(w) \neq 0$, the total score, the maximal score, and the score of the last token in the tweet. We also use *The MPQA Subjectivity Lexicon* (Wilson et al., 2005), *Bing Liu's Opinion Lexicon* (Ding et al., 2008), and the *NRC Emotion Lexicon* (Mohammad and Turney, 2010), assigning scores of $+/-2$ for strong and $+/-1$ for weak degrees of sentiment. The resulting four feature vectors contain the sum of positive and negative scores for tokens in affirmative and negated contexts, equivalently to Kiritchenko et al. (2014).

| Feature | Description |
|---|---|
| Word $n$-grams | contiguous token sequences |
| Char $n$-grams | contiguous character sequences |
| Negated tokens | number of negated tokens |
| Sentiment lexica | feature set for each lexicon |
| Clusters | tokens from '1000 word clusters' |
| POS | part-of-speech tag frequency |
| All caps | upper-case tokens |
| Elongated | tokens with repeated characters |
| Emoticons | positive and negative emoticons |
| Punctuation | punctuation mark sequences |
| Hashtags | number of hashtags |

Table 6: Sentiment classifier STATE feature set

Instead of adding only the presence of words from each of the 1000 clusters from CMU's Tweet NLP tool[3] in the *clusters* feature, as Kiritchenko et al. (2014) did, we count occurrences for each cluster and represent them with a feature. Input to the *POS* feature is obtained from the Twitter part-of-speech tagger (Owoputi et al., 2013). The *emoticons* feature is the number of happy and sad emoticons, and whether a tweet's last token is happy or a sad. The *all-caps*, *elongated* (tokens with characters repeated more than two times), *punctuation* (exclamation or question marks), and *hashtag* features are straight-forward counts of the number of tokens of each type. All the matrices from the different parts of the feature extraction are concatenated column-wise into the final feature matrix, and scaled in order to be suitable as input to a classifier.

The classifier step declares which classifier to use, along with its default parameters. It is passed the resulting feature matrix from the feature extraction, with which it creates the decision space if training, or classifies samples if predicting. Using the negation scope classifier with the parameters identified in Section 5.3, a grid search was performed over the entire Twitter2013-train data set using stratified 10-fold cross validation to find the $C$ and $\gamma$ parameters for the SVM classifier. A preliminary coarse search showed the radial basis function (RBF) kernel to yield the best results, although most state-of-the-art sentiment classification systems use a linear kernel.

A finer parameter space was then examined. The surface plots in Figure 1 display the effects of the $C$

[2] `mathiasbynens.be/demo/url-regex`

[3] `http://www.ark.cs.cmu.edu/TweetNLP/`

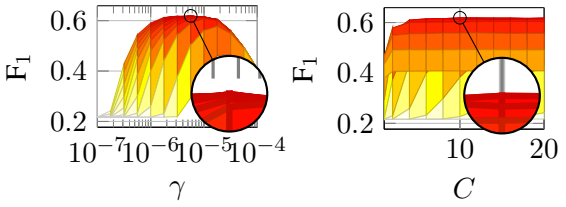

Figure 1: SVM grid search $F_1$ scores for $\gamma$ and $C$

and $\gamma$ parameters on the classifier's $F_1$ score. The combination of parameters that scored best was $C = 10$ and $\gamma \approx 5.6 * 10^{-6}$, marked by circles. Increasing $C$ beyond 10 gives no notable change in $F_1$ score. The combination of a small $\gamma$ and higher values of $C$ means that the classifier is quite generalized, and that increasing $C$ (regularizing further) makes no difference. It also suggests that the data is noisy, requiring a great deal of generalization.

In order to allow a user to query Twitter for a search phrase on live data, the classifier is wrapped in a web application using the Django web framework.[4] The resulting tweet hits are classified using a pre-trained classifier, and presented to the user indicating their sentiment polarities. The total distribution of polarity is also displayed as a graph to give the user an impression of the overall opinion.

### 6.2 Sentiment Classifier Performance

The SVM was trained on the Twitter2013-train set using the parameters identified through grid search, and tested on the Twitter2014-test and Twitter2013-test sets, scoring as in Table 7. Sentiment classification is here treated as a three-class task, with the labels positive, negative, and objective/neutral. In addition to precision, recall, and $F_1$ for each class, we report the *macro-average* of each metric across all classes. Macro-averaging disregards class imbalance and is calculated by taking the average of the classification metric outputs for each label, equally weighting each label, regardless of its number of samples. The last column of the table shows the *support*: the number of samples for each label in the test set.

As can be seen in the table, the classifier performed worst on negative samples. Figure 2 displays the confusion matrices for the Twitter2013-test set (the Twitter2014 matrices look similar). If there were perfect correlation between true and predicted labels, the diagonals would be completely red. However, the confusion matrices show (clearer in the normalized version) that the classifier is quite biased towards the neutral label (illustrated with 😐),

[4] https://djangoproject.com

| Label | P | R | $F_1$ | Support |
|---|---|---|---|---|
| | *Twitter2014-test* | | | |
| positive | 0.863 | 0.589 | 0.700 | 805 |
| neutral | 0.568 | 0.872 | 0.688 | 572 |
| negative | 0.717 | 0.487 | 0.580 | 156 |
| avg / total | 0.738 | 0.684 | 0.684 | 1533 |
| | *Twitter2013-test* | | | |
| positive | 0.851 | 0.581 | 0.691 | 1273 |
| neutral | 0.627 | 0.898 | 0.739 | 1369 |
| negative | 0.711 | 0.426 | 0.533 | 467 |
| avg / total | 0.731 | 0.697 | 0.688 | 3109 |

Table 7: Sentiment classifier performance

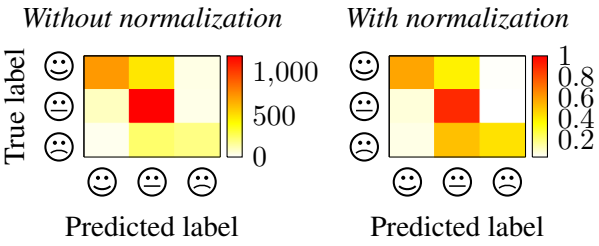

Figure 2: Sentiment classifier confusion matrices

as can be seen from the warm colours in the 😊 and ☹ true label cells of the 😐 predicted label column, in particular misclassifying negative samples. This is likely an effect of the imbalanced training set, where neutral samples greatly outnumber negative.

### 6.3 TSA Feature Ablation Study

The results of an ablation study of the TSA classifier are shown in Table 8, where the *all* rows ($n$-grams/counts) refer to removing all features in that group. Most apparently, the *sentiment lexica* feature has the greatest impact on classifier performance, especially on the Twitter2013-test set. This may be since the most important lexica (Sentiment140 and NRC Hashtag Sentiment) were created at the same time as the Twitter2013 data, and could be more accurate on the language used then.

The *character $n$-gram* feature slightly damages performance on the Twitter2014-test set, although making a positive contribution on the Twitter2013 data. This is most likely caused by noise in the data, but the feature could be sensitive to certain details that appeared after the Twitter2013 data collection.

The majority of the count features do not impose considerable changes in performance, although the *all-caps* feature decreases performance on both test data sets, most likely only introducing noise.

| | Features | Twitter test 2014 | Twitter test 2013 |
|---|---|---|---|
| | All | 0.684 | 0.688 |
| n-grams | − word $n$-grams | 0.672 | 0.674 |
| | − char $n$-grams | 0.688 | 0.676 |
| | − *all $n$-grams* | 0.664 | 0.667 |
| | − sentiment lexica | 0.665 | 0.657 |
| frequency count features | − clusters | 0.666 | 0.677 |
| | − POS | 0.684 | 0.685 |
| | − all caps | 0.685 | 0.689 |
| | − elongated | 0.682 | 0.687 |
| | − emoticons | 0.681 | 0.688 |
| | − punctuation | 0.682 | 0.688 |
| | − hashtag | 0.684 | 0.688 |
| | − negation | 0.684 | 0.688 |
| | − *all counts* | 0.665 | 0.671 |

Table 8: Sentiment classifier ablation ($F_1$ scores)

However, the Tweet NLP *clusters* feature has a large impact, as anticipated. Tweets contain many misspellings and unusual abbreviations and expressions, and the purpose of this feature is to make generalizations by counting the occurrences of clusters that include similar words.

### 6.4 Effect of Negation Scope Detection

Table 9 shows the effects of performing negation scope detection on several variations of the sentiment classification system and data sets. The first six rows give results from experiments using the Twitter2013-training and Twitter2014-test sets, and the remaining rows results when using only a subset of the data: tweets that contain negation, as determined by our NSD system. The rows are grouped into four segments, where each segment shows scores for a classifier using either no, simple or sophisticated negation scope detection. The segments represent different feature sets, either using *all* features or only the features that are directly affected by *negation*: word and character $n$-grams, sentiment lexica, and negation counts.

In every case, taking negation into account using either the simple or the sophisticated method improves the $F_1$ score considerably. Using all the data, the sophisticated solution scores marginally better than the simple one, but it improves more clearly upon the simple method on the negated part of the data, with $F_1$ improvements ranging from $4.5\,\%$ to $6.1\,\%$ (i.e., from 0.029 to 0.039 $F_1$ score).

| Features | NSD method | P | R | $F_1$ |
|---|---|---|---|---|
| *All tweets (training and test sets)* | | | | |
| all | No | 0.730 | 0.659 | 0.653 |
| | Simple | 0.738 | 0.676 | 0.675 |
| | Sophisticated | 0.738 | 0.684 | 0.684 |
| negation | No | 0.705 | 0.618 | 0.601 |
| | Simple | 0.728 | 0.663 | 0.662 |
| | Sophisticated | 0.729 | 0.667 | 0.665 |
| *Only tweets containing negation* | | | | |
| all | No | 0.598 | 0.599 | 0.585 |
| | Simple | 0.653 | 0.654 | 0.644 |
| | Sophisticated | 0.675 | 0.682 | 0.673 |
| negation | No | 0.609 | 0.604 | 0.586 |
| | Simple | 0.648 | 0.654 | 0.633 |
| | Sophisticated | 0.681 | 0.696 | 0.672 |

Table 9: Sentiment classification results

## 7 Conclusion and Future Work

The paper has introduced a sophisticated approach to negation scope detection (NSD) for Twitter sentiment analysis. The system consists of two parts: a negation cue detector and a negation scope classifier. The cue detector uses a lexicon lookup that yields high recall, but modest precision. However, the negation scope classifier still produces better results than observed in other domains: an $F_1$ score of $0.853$ with $64.5\,\%$ correctly classified scopes, indicating that the Conditional Random Fields-based scope classifier is able to identify the trend of certain dictionary cues being misclassified.

A sentiment classifier for Twitter data was also developed, incorporating several features that benefit from negation scope detection. The results confirm that taking negation into account in general improves sentiment classification performance significantly, and that using a sophisticated NSD system slightly improves the performance further.

The negation cue variation in the Twitter data was quite low, but due to part-of-speech ambiguity it was for some tokens unclear whether or not they functioned as a negation signal. A more intricate cue detector could in the future aim to resolve this.

The study builds on current state-of-the-art Twitter sentiment analysis features, but other features could tentatively make better use of well-performing negation scope detection. The negated contexts underlying the utilized sentiment lexica are, for example, based on a simple NSD model, so might be improved by more elaborate solutions.

## References


Apoorv Agarwal, Boyi Xie, Ilia Vovsha, Owen Rambow, and Rebecca Passonneau. 2011. Sentiment analysis of Twitter data. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 30–38, Portland, Oregon, June. ACL. Workshop on Languages in Social Media.

Héctor Cerezo-Costas and Diego Celix-Salgado. 2015. Gradiant-analytics: Training polarity shifters with CRFs for message level polarity detection. In *Proceedings of the 9th International Workshop on Semantic Evaluation*, pages 539–544, Denver, Colorado, June. ACL.

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

Wendy W. Chapman, Will Bridewell, Paul Hanbury, Gregory F. Cooper, and Bruce G. Buchanan. 2001. A simple algorithm for identifying negated findings and diseases in discharge summaries. *Journal of Biomedical Informatics*, 34(5):301–310, October.

Isaac G. Councill, Ryan McDonald, and Leonid Velikovich. 2010. What's great and what's not: learning to classify the scope of negation for improved sentiment analysis. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 51–59, Uppsala, Sweden, July. ACL. Workshop on Negation and Speculation in Natural Language Processing.

Xiaowen Ding, Bing Liu, and Philip S. Yu. 2008. A holistic lexicon-based approach to opinion mining. In *Proceedings of the 2008 International Conference on Web Search and Data Mining*, pages 231–240, Stanford, California, February. ACM.

Richárd Farkas, Veronika Vincze, György Móra, János Csirik, and György Szarvas. 2010. The CoNLL-2010 shared task: Learning to detect hedges and their scope in natural language text. In *Proceedings of the 14th Conference on Computational Natural Language Learning*, pages 1–12, Uppsala, Sweden, July. ACL.

Talmy Givón. 1993. *English grammar: A function-based introduction*. John Benjamins, Amsterdam, The Netherlands.

Tobias Günther, Jean Vancoppenolle, and Richard Johansson. 2014. RTRGO: Enhancing the GU-MLT-LT system for sentiment analysis of short messages. In *Proceedings of the 8th International Workshop on Semantic Evaluation*, pages 497–502, Dublin, Ireland, August. ACL.

Matthias Hagen, Martin Potthast, Michael Büchner, and Benno Stein. 2015. Webis: An ensemble for Twitter sentiment detection. In *Proceedings of the 9th International Workshop on Semantic Evaluation*, pages 582–589, Denver, Colorado, June. ACL.

Hussam Hamdan, Patrice Bellot, and Frederic Bechet. 2015. Lsislif: Feature extraction and label weighting for sentiment analysis in Twitter. In *Proceedings of the 9th International Workshop on Semantic Evaluation*, pages 568–573, Denver, Colorado, June. ACL.

Svetlana Kiritchenko, Xiaodan Zhu, and Saif M. Mohammad. 2014. Sentiment analysis of short informal texts. *Journal of Artificial Intelligence Research*, 50:723–762, August.

Lingpeng Kong, Nathan Schneider, Swabha Swayamdipta, Archna Bhatia, Chris Dyer, and Noah A. Smith. 2014. A dependency parser for tweets. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1001–1012, Doha, Qatar, October. ACL.

Natalia Konstantinova, Sheila C.M. de Sousa, Noa P. Cruz, Manuel J. Maña, Maite Taboada, and Ruslan Mitkov. 2012. A review corpus annotated for negation, speculation and their scope. In *Proceedings of the 8th International Conference on Language Resources and Evaluation*, pages 3190–3195, Istanbul, Turkey, May. ELRA.

Yasuhide Miura, Shigeyuki Sakaki, Keigo Hattori, and Tomoko Ohkuma. 2014. TeamX: A sentiment analyzer with enhanced lexicon mapping and weighting scheme for unbalanced data. In *Proceedings of the 8th International Workshop on Semantic Evaluation*, pages 628–632, Dublin, Ireland, August. ACL.

Saif Mohammad and Peter Turney. 2010. Emotions evoked by common words and phrases: Using Mechanical Turk to create an emotion lexicon. In *Proceedings of the 2010 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 26–34, Los Angeles, California, June. ACL. Workshop on Computational Approaches to Analysis and Generation of Emotion in Text.

Saif Mohammad, Svetlana Kiritchenko, and Xiaodan Zhu. 2013. NRC-Canada: Building the state-of-the-art in sentiment analysis of tweets. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the 7th International Workshop on Semantic Evaluation*, SemEval '13, pages 321–327, Atlanta, Georgia, June. ACL.

Saif Mohammad. 2012. #Emotional tweets. In *First Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the 6th International Workshop on Semantic Evaluation*, pages 246–255, Montréal, Canada, June. ACL.

Karo Moilanen and Stephen Pulman. 2007. Sentiment composition. In *Proceedings of the 6th International Conference on Recent Advances in Natural Language Processing*, pages 378–382, Borovets, Bulgaria, September.

Roser Morante and Eduardo Blanco. 2012. * SEM 2012 shared task: Resolving the scope and focus of negation. In *First Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the 6th International Workshop on Semantic Evaluation*, pages 265–274, Montréal, Canada, June. ACL.

Roser Morante and Walter Daelemans. 2009. A metalearning approach to processing the scope of negation. In *Proceedings of the 13th Conference on Computational Natural Language Learning*, pages 21–29, Boulder, Colorado, June. ACL.

Roser Morante and Caroline Sporleder. 2012. Modality and negation: An introduction to the special issue. *Computational Linguistics*, 38(2):223–260.

Pradeep G. Mutalik, Aniruddha Deshpande, and Prakash M. Nadkarni. 2001. Use of general-purpose negation detection to augment concept indexing of medical documents: A quantitative study using the UMLS. *Journal of the American Medical Informatics Association*, 8(6):598–609, November.

Preslav Nakov, Zornitsa Kozareva, Sara Rosenthal, Veselin Stoyanov, Alan Ritter, and Theresa Wilson. 2013. SemEval-2013 Task 2: Sentiment analysis in Twitter. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the 7th International Workshop on Semantic Evaluation*, SemEval '13, pages 312–320, Atlanta, Georgia, June. ACL.

Joakim Nivre, Johan Hall, Jens Nilsson, Atanas Chanev, Gülşen Eryiğit, Sandra Kübler, Svetoslav Marinov, and Erwin Marsi. 2007. MaltParser: A language-independent system for data-driven dependency parsing. *Natural Language Engineering*, 13(2):95–135, January.

Naoaki Okazaki. 2007. CRFsuite: a fast implementation of conditional random fields (CRFs). `www.chokkan.org/software/crfsuite/`.

Olutobi Owoputi, Brendan O'Connor, Chris Dyer, Kevin Gimpel, Nathan Schneider, and Noah A. Smith. 2013. Improved part-of-speech tagging for online conversational text with word clusters. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 380–390, Atlanta, Georgia, June. ACL.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12(1):2825–2830.

Terry Peng and Mikhail Korobov. 2014. python-crfsuite. `github.com/tpeng/python-crfsuite`.

Nataliia Plotnikova, Micha Kohl, Kevin Volkert, Stefan Evert, Andreas Lerner, Natalie Dykes, and Heiko Ermer. 2015. KLUEless: Polarity classification and association. In *Proceedings of the 9th International Workshop on Semantic Evaluation*, pages 619–625, Denver, Colorado, June. ACL.

Christopher Potts. 2011. Sentiment symposium tutorial. In *Sentiment Analysis Symposium*, San Francisco, California, November. Alta Plana Corporation. `http://sentiment.christopherpotts.net/`.

Sara Rosenthal, Alan Ritter, Preslav Nakov, and Veselin Stoyanov. 2014. SemEval-2014 Task 9: Sentiment analysis in Twitter. In *Proceedings of the 8th International Workshop on Semantic Evaluation*, pages 73–80, Dublin, Ireland, August. ACL.

Sara Rosenthal, Preslav Nakov, Svetlana Kiritchenko, Saif Mohammad, Alan Ritter, and Veselin Stoyanov. 2015. SemEval-2015 Task 10: Sentiment analysis in Twitter. In *Proceedings of the 9th International Workshop on Semantic Evaluation*, pages 451–463, Denver, Colorado, June. ACL.

Øyvind Selmer, Mikael Brevik, Björn Gambäck, and Lars Bungum. 2013. NTNU: Domain semi-independent short message sentiment classification. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the 7th International Workshop on Semantic Evaluation*, SemEval '13, pages 430–437, Atlanta, Georgia, June. ACL.

Duyu Tang, Furu Wei, Bing Qin, Ting Liu, and Ming Zhou. 2014. Coooolll: A deep learning system for Twitter sentiment classification. In *Proceedings of the 8th International Workshop on Semantic Evaluation*, pages 208–212, Dublin, Ireland, August. ACL.

Gunnel Tottie. 1991. *Negation in English Speech and Writing: A Study in Variation*. Academic Press, San Diego, California.

Veronika Vincze, György Szarvas, Richárd Farkas, György Móra, and János Csirik. 2008. The BioScope corpus: biomedical texts annotated for uncertainty, negation and their scopes. *BMC Bioinformatics*, 9(Suppl 11):S9, November.

Michael Wiegand, Alexandra Balahur, Benjamin Roth, Dietrich Klakow, and Andrés Montoyo. 2010. A survey on the role of negation in sentiment analysis. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 60–68, Uppsala, Sweden, July. ACL. Workshop on Negation and Speculation in Natural Language Processing.

Theresa Wilson, Paul Hoffmann, Swapna Somasundaran, Jason Kessler, Janyce Wiebe, Yejin Choi, Claire Cardie, Ellen Riloff, and Siddharth Patwardhan. 2005. OpinionFinder: A system for subjectivity analysis. In *Proceedings of the 2005 Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 34–35, Vancouver, British Columbia, Canada, October. ACL. Demonstration Abstracts.

Qiaoming Zhu, Junhui Li, Hongling Wang, and Guodong Zhou. 2010. A unified framework for scope learning via simplified shallow semantic parsing. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 714–724, Cambridge, Massachusetts, October. ACL.