A time-series classification model for Twitter opinion lexicon expansion

Felipe Bravo-Marquez
Department of Computer Science, University of Waikato, New Zealand
fjb11@students.waikato.ac.nz

Eibe Frank
Department of Computer Science, University of Waikato, New Zealand
eibe@waikato.ac.nz

Bernhard Pfahringer
Department of Computer Science, University of Waikato, New Zealand
bernhard@waikato.ac.nz

ABSTRACT

Opinion lexicons, which are lists of terms labelled by sentiment, are widely used resources to support automatic sentiment analysis of textual passages. However, existing resources of this type exhibit some limitations when applied to social media messages such as tweets (posts in Twitter), because they are unable to capture the diversity of informal expressions commonly found in this type of media. In this article, we propose a process to automatically categorise Twitter words into three sentiment classes to perform lexicon expansion: positive, neutral, and negative. This is done through a supervised learning process in which word-level attributes are calculated from a multi-dimensional time-series, and a seed lexicon of labelled words is used as the training dataset. The time-series are extracted from a temporal collection of automatically labelled tweets representing the relationship between a word and the sentiment expressed in the tweets where it occurs. Our experimental results show that our method outperforms the word-level sentiment classification performance obtained by semantic orientation, a state-of-the-art measure for establishing world-level sentiment. Additionally, we show that our expanded lexicon produces significant improvements when used for message-level polarity classification of tweets.

Categories and Subject Descriptors

I.2.7.7 [**Artificial Intelligence**]: Natural Language Processing—*Text Analysis*

General Terms

Experimentation, Measurement

Keywords

Lexicon Expansion, Sentiment Analysis, Twitter

1. INTRODUCTION

Many sentiment analysis methods rely on lexical resources for evaluating the sentiment of a text passage, in particular they often rely on an opinion lexicon. An opinion or sentiment lexicon is a dictionary of opinion words with their corresponding sentiment values. Lexicons can be used to compute the polarity of a message by aggregating the orientation values of the opinion words it contains [10]. They have also proven to be useful when used to extract features in supervised classification schemes [12, 15, 32].

Social media platforms and, in particular, microblogging services such as **Twitter**[1], are increasingly being adopted by people to access and publish information about a great variety of topics. Sentiment analysis applied to social media posts has received increasing interest due to its importance in a wide range of fields such as business, sports, and politics. However, the language used in Twitter provides substantial challenges for sentiment analysis. The words used in this platform include many abbreviations, acronyms, and misspelled words that are not observed in traditional media and are not covered by popular lexicons such as OpinionFinder [29], SentiWordnet [6], and the Harvard General Inquirer [24]. The diversity and sparseness of these informal words make the manual creation of a Twitter-oriented opinion lexicon a time-consuming task.

In this article we propose a supervised framework for opinion lexicon expansion for the language used in Twitter. Taking SentiWordnet as inspiration, each word in our expanded lexicon has a probability distribution, describing how positive, negative, and neutral it is. Additionally, all the entries of the lexicon are associated with a corresponding part-of-speech tag. We believe that estimating the sentiment distribution of POS-tagged words may be useful for developing Twitter-specific sentiment applications for the following reasons:

1. A word can present certain levels of intensity [25] for a specific sentiment category e.g., the word *awesome* is more positive than the word *adequate*. The estimated probabilities can be used to represent these levels of intensities.

2. The neutral score provided by the lexicon is useful for discarding non-opinion words in text-level polarity classification tasks. This can be easily done by discarding words classified as neutral. On the other hand, unsupervised lexicon expansion techniques such as semantic orientation [26] provide a single numerical score for each word. It is not clear how to find a threshold for using this score in neutrality detection.

3. Homographs, which are words that share the same spelling but have different meanings, should have different lexicon entries for each different meaning. By relying on POS-tagged words, homographs with different POS-tags will be disambiguated [28]. For instance, the word *apple* will receive dif-

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

ferent sentiment scores when used to refer to a common noun (a fruit) or a proper noun (a company).

The proposed methodology takes a collection of tweets labelled according to their polarity to establish a polarity-related time-series for each POS-tagged word in the tweets and uses features extracted from this time-series in relation with the labels provided by a seed lexicon to train a word-level sentiment classifier. The seed lexicon is taken from the union of four different hand-made lexicons after discarding all polarity clashes from the intersection. Furthermore, the tweets from the collection are labelled using a semi-supervised heuristics called *emoticon-based annotation* to avoid the expensive costs of data annotation. In this approach, only tweets with positive or negative emoticons are considered and labelled according to the polarity indicated by the emoticon. This idea has been widely used before to train message-level sentiment classifiers [3, 8, 22].

Our word-level time-series are computed from two different criteria: Semantic Orientation (SO) [26], which is based on the mutual information between word and sentiment class, and Stochastic Gradient Descent (SGD) score, which learns a linear relationship between word and sentiment class.

The proposed procedure can be summarised in the following steps:

1. Collect tweets from the domain and the time period for which the lexicon needs to be expanded.
2. Label the collection with sentiment classes in an automatic way.
3. Tag all the words using a part-of-speech tagger.
4. Calculate word-level time-series for all tagged words and extract word-level sentiment features from them.
5. Label those words with a sentiment that match an existing hand-made polarity lexicon.
6. Train a word-level classifier using the word-level features and the words labels from the seed lexicon.
7. Use the trained classifier to estimate the polarity distribution of the remaining unlabelled words.

As will be discussed in Section 2, the automatic expansion of Twitter opinion words has been studied before [1, 18, 31]. In all these works, the expansion was conducted using semantic orientation, which is an unsupervised measure. To the best of our knowledge, this is the first article in which the lexicon expansion of Twitter opinion words is studied and evaluated using POS disambiguation and supervised learning. Additionally, this is the first study in which scores for positive, negative, and neutral opinion-related properties are provided for Twitter-specific expressions.

We test our approach on two collections of automatically labelled tweets. Our results indicate that our supervised framework outperforms the classification accuracy obtained by semantic orientation when the detection of neutral words is included. We also evaluate the usefulness of the expanded lexicon for message-level polarity classification of tweets, showing significant improvements in performance.

This article in organised as follows. In Section 2 we provide a review of existing work in opinion lexicon expansion. In Section 3 we describe the seed lexicon used to label the words for the training set. In Section 4 we explain the mechanisms studied to automatically create collections of labelled tweets. The creation of our word-level time-series is described in Section 5, together with the features used for training the classifier. In Section 6 we present the experiments we conducted to evaluate the proposed approach and we also discuss the obtained results. The main findings and conclusions are discussed in Section 7.

2. RELATED WORK

There are basically two type of resources which can be exploited for lexicon expansion: a thesaurus and a corpus of documents. The simplest approach using a thesaurus such as WordNet[2] is to expand a seed lexicon of labelled opinion words using synonyms and antonyms from the lexical relations provided by the thesaurus [11, 14]. The hypothesis behind this approach is that synonyms have the same polarity and antonyms have the opposite. This process is normally iterated several times. In [13] a graph was created using WordNet adjectives as vertices and the synonym relation as edges. The orientation of a term is determined by its relative distance from the two seed terms *good* and *bad*. In [5] a supervised classifier was trained using a seed of labelled words which was obtained through synonyms and antonyms expansion. For each word, a vector space model is created from the definition or *gloss* provided by the WordNet dictionary. This representation is used to train a word-level classifier that is used for lexicon expansion. An equivalent approach was applied later to create SentiWordnet[3] in [6, 2]. In SentiWordNet each WordNet *synset* or group of synonyms is annotated into classes positive, negative and neutral in the range $[0, 1]$.

A limitation of thesaurus-based approaches is their inability to capture domain dependent words. Corpus-based approaches exploit syntactic or co-occurrence patterns to expand the lexicon to the words found within a collection of documents. In [9], the proposed method starts with a set of adjectives whose polarity is known and then discovers the polarities of new adjectives using some linguistic patterns from a corpus of documents. The authors show, using log-linear regression, that conjunctions between adjectives provide indirect information about the orientation. For example, adjectives connected with the conjunction "and" tend to have the same orientation and adjectives connected with the conjunction "but" tend to have opposite orientation. This approach allows for the extraction of domain-dependent information and the adaptation to new domains when the corpus of documents is changed.

In [26, 27], the expansion is done through a measure referred to as *semantic orientation* which is based on the the point-wise mutual information (PMI) between two random variables:

$$\mathrm{PMI}(term_1, term_2) = \log_2 \left(\frac{Pr(term_1 \wedge term_2)}{Pr(term_1) Pr(term_2)} \right) \quad (1)$$

The semantic orientation of a word is the difference between the PMI of the word with a positive emotion and a negative emotion. Different ways have been proposed to represent the joint probabilities of words and emotions. In Turney's work [26] they are estimated using the number of hits returned by a search engine in response to a query composed of the target word together with the word "excellent" and another query using the word "poor" in the same way.

The same idea was used for Twitter lexicon expansion in [1, 18, 31]. All these works model the joint probabilities from collections of tweets labelled in automatic ways. In [1] the tweets are labelled from a trained classifier using thresholds for the different classes to ensure high precision. In [31], they are labelled from emoticons to create domain-specific lexicons. In [18], they are labelled from

[2] http://wordnet.princeton.edu/

[3] http://sentiwordnet.isti.cnr.it/

emoticons and hashtags associated with emotions to create two different lexicons. These lexicons were tested for tweet-level polarity classification.

A more detailed survey on sentiment lexicon expansion is provided in the book by Liu [16], in Chapter 6.

3. GROUND-TRUTH WORD POLARITIES

In this section, we describe the seed lexicon used to label the training dataset for our word sentiment classifier. There are two main properties that vary from one lexicon to another: the way in which the lexicon is built and the nature of the sentiment label. Lexicons can be created manually or automatically. Manually created lexicons tend to be smaller and less noisy than automatically made lexicons [4].

The labels can be categorical, with classes -positive, negative, and neutral- (in some cases neutral words are not considered), or numerical, representing the strength of the polarity (e.g., from -5 to 5). The labels can also express additional emotional states.

As we need to reduce the noise in our training data, we consider the following hand-made lexicons for data labelling:

- *MPQA Subjectivity Lexicon*: This lexicon was created by Wilson et al. [29] and is part of OpinionFinder[4], a system that automatically detects subjective sentences in document corpora. The lexicon has positive, negative, and neutral words.
- *Bing Liu*: This lexicon is maintained and distributed by Bing Liu[5] and was used in several papers authored or co-authored by him [16]. The lexicon has positive and negative entries.
- *Afinn*: This strength-oriented lexicon [20] has positive words scored from 1 to 5 and negative words scored from -1 to -5. It includes slang and obscene words and also acronyms and Web jargon. We tagged words with negative and positive scores to negative and positive classes respectively.
- *NRC emotion Lexicon*: This emotion-oriented lexicon was created by Mohammad and Turney [17] by conducting a tagging process on the crowdsourcing Amazon Mechanical Turk platform. In this lexicon, the words are annotated according to eight emotions: joy, trust, sadness, anger, surprise, fear, anticipation, and disgust, and two polarity classes: positive and negative. There are many words that are not associated with any emotional state which were tagged as neutral. In this work, we consider positive, negative, and neutral tags from this lexicon.

We create a meta-lexicon by taking the union of these resources and discarding all words where a polarity clash is observed. A polarity clash is a word that receives two or more different tags in the union of lexicons. The number of words for the different polarity classes in the different lexicons is displayed in Table 1.

From the table we can observe that the number of words per class is significantly reduced after removing the clashes from the union. The total number of clashes is 1074 and a sample of them in presented in Figure 1.

This high number of clashes found among different hand-made lexicons indicates two things: 1) Different human annotators can disagree when tagging a word to polarity classes. 2) There are several words that can belong to more than one sentiment class. Due to this, we can say that word-level polarity classification is a hard and subjective problem.

[4] http://mpqa.cs.pitt.edu/opinionfinder/opinionfinder_2/

[5] http://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html

	Positive	Negative	Neutral
AFINN	564	964	0
Bing Liu	2003	4782	0
MPQA	2295	4148	424
NRC-Emo	2312	3324	7714
Meta-Lexicon	3730	6368	7088

Table 1: Lexicon Statistics

Figure 1: Polarity clashes

4. OBTAINING LABELLED TWEETS

In order to calculate our word-level time-series, we require a collection of time-stamped tweets with their corresponding polarity labels.

We rely on the emoticon-based annotation approach in which tweets exhibiting positive :) and negative :(emoticons are labelled according to the emoticon's polarity [8]. Afterwards, the emoticon used to label the passage is removed from the content.

The word polarities of the expanded lexicon will reflect the domains found in the collection from which the lexicon is built. In this work, with the aim of creating general purpose lexicons, we consider two collections of tweets covering multiple topics: The Edinburgh corpus (ED) [23], and the Stanford Sentiment corpus (STS) [8]. These collections were gathered from two Twitter APIs: the streaming API[6], from which a real time sample of public posts can be retrieved, and the search API[7], which allows the submission of queries composed of key-terms.

The ED corpus has 97 million tweets which were collected using the Twitter streaming API from a period spanning November 11th 2009 until February 1st 2010. This collection includes tweets in multiple languages. As was done in [3], non-English tweets were filtered out, and tweets exhibiting positive or negative emoticons were annotated using the emoticon-based approach. The remaining tweets were discarded.

The STS corpus was created by periodically sending queries :) and :(to the Twitter search API between April 6th 2009 to June 25th 2009. All the tweets in this collection are written in English. The collection was labelled using the emoticon-based annotation approach.

The number of tweets for each polarity class in the two corpora is given in Table 2. We can observe that when using the streaming API (ED) positive emoticons occur much more frequently than negative ones.

[6] https://dev.twitter.com/streaming/overview

[7] https://dev.twitter.com/rest/public/search

	ED	STS
Positive	1,813,705	800,000
Negative	324,917	800,000
Total	2,138,622	1,600,000

Table 2: Collection statistics

5. WORD-LEVEL FEATURES

The word-level features proposed in this work exploit the temporal structure of a collection of polarity-labelled tweets. All the tweets from the collection are lowercased, tokenised and POS-tagged. We use the TwitterNLP library [7] that provides a tokeniser and a tagger specific for the language used in Twitter. The tagger includes nominal words classes, open and closed word classes, and Twitter specific classes such as hashtags, user mentions and emoticons. We prepend a POS-tag prefix to each word in order to differentiate homographs exhibiting different POS-tags.

We treat the time-sorted collection of tweets as a data stream and create two types of time-series for each POS-tagged word observed in the vocabulary: the SGD series, and the SO series.

The first time-series is calculated by incrementally training a linear support vector machine using a stochastic gradient descent (SGD) online learning process [30]. The weights of this linear model correspond to POS-tagged words that are updated in an incremental fashion. We optimise the hinge loss function with an L_2 penalty and a learning rate equal to 0.1:

$$\frac{\lambda}{2}||w||^2 + \sum[1 - y(\mathbf{x}\mathbf{w} + b)]_+. \quad (2)$$

The variables $\mathbf{w}$, b, and λ correspond to the weight vector, the bias, and the regularisation parameter, respectively. The class labels y are assumed to be in $\{+1, -1\}$, corresponding to positively and negatively labelled tweets, respectively. The regularisation parameter was set to 0.0001. As was suggested in [3], the model's weights determine how strongly the presence of a word influences the prediction of negative and positive classes. The SGD time-series is created by applying this learning process to a collection of labelled tweets and storing the word's coefficients in different time windows. We use time windows of 1,000 examples.

The second time-series corresponds to the accumulated semantic orientation (SO) introduced in Section 2. Let *count* be a function that counts the number of times that a word or a sentiment label has been observed during a certain period of time. We calculate the SO score for each POS-tagged word in an accumulated way according to the following expression:

$$\mathrm{SO}(word) = log_2\left(\frac{\mathrm{count}(\mathrm{word} \wedge \mathrm{pos}) \times \mathrm{count}(\mathrm{neg})}{\mathrm{count}(\mathrm{word} \wedge \mathrm{neg}) \times \mathrm{count}(\mathrm{pos})}\right) \quad (3)$$

We use time windows of 1,000 examples and the Laplace correction to avoid the zero-frequency problem.

We create SGD and SO time-series for positive and negative labelled tweets in our two datasets: ED, and STS. We refer to these time-series as **sgd.posneg** and **so.posneg**.

We rely on our time-series to extract features that are used to train our world-level polarity classifier. These features summarise properties of location and dispersion of the time-series and are listed in Table 3. Location-oriented features *mean*, *trimm.mean* and *median* measure the central tendency of the time-series. The feature *last.element* corresponds to the last value observed in the time-series. This attribute would be equivalent to the traditional SO measure for the so.posneg time-series. The features *sd*, *iqr*, *sg*, and *sg.diff* measure the level of dispersion of the time-series.

In addition to these time-series features, we include the POS-tag of the word as a nominal attribute.

Feature	Description
mean	The mean of the time-series.
trunc.mean	The truncated mean of the time-series.
median	The median of the time-series
last.element	The last observation of the time-series.
sd	The standard deviation of the time-series .
iqr	The inter-quartile range.
sg	The fraction of times the time-series changes its sign.
sg.diff	The sg value for the differenced time-series.

Table 3: Time-series features

6. EXPERIMENTS

In this section, we present our experimental results for Twitter lexicon expansion. In the first part, we study the word-level polarity classification problem. In the second part, we expand the lexicon using the trained classifier and use it for message-level polarity classification of tweets.

6.1 Word-level polarity classification

We calculated the time-series described in Section 5 for the most frequent 10,000 POS-tagged words found in each of our two datasets. The time-series were calculated using MOA[8], a data stream mining framework. The resulting time-series **sgd.posneg** and **so.posneg** for a sample of words in the STS dataset are shown in Figure 2.

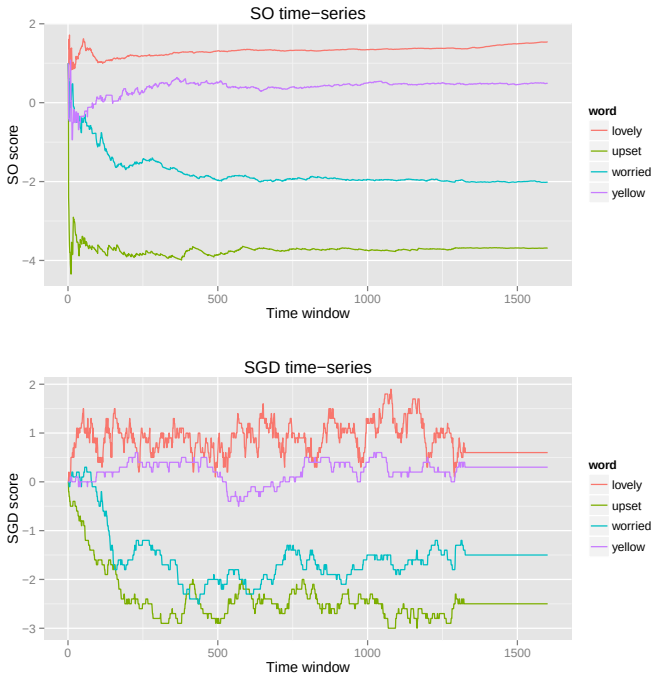

Figure 2: Word-level time-series.

[8] http://moa.cs.waikato.ac.nz/

	ED	STS
Positive	1027	1023
Negative	806	985
Neutral	1814	1912
Total	3647	3920

Table 4: Word-level polarity classification datasets.

We can observe that the positive and neutral words *lovely* and *yellow* exhibit greater SO and SGD scores than the negative words *worried* and *upset*. This suggests a correspondence between time-series values and the word polarities. However, we can see that the SO time-series are much more stable than the SGD ones. We believe that SO is more stable than SGD because, as shown in Equation 3, SO treats words independently from each other given the sentiment class. On the other hand, SGD scores are updated according to the SGD learning rule, which comes from the sub-gradient of Equation 2. In this rule, the coefficients are updated every time the learning SVM misclassifies an example from the stream of emoticon-labelled tweets ($y(\mathbf{x}\mathbf{w} + b) < 1$). Therefore, the change of the weight of a particular word depends both on the sentiment label, and the co-occurring words within the tweets from the training collection.

To create training and test data for machine learning, all the POS-tagged words matching the metalexicon are labelled according to the lexicon's polarities. It is interesting to consider how frequently positive, negative, and neutral words occur in a collection of tweets. The number of words labelled as positive, negative, and neutral for both the ED and STS datasets is given in Table 4. As shown in the table, neutral words are the most frequent words in both datasets. Moreover, positive words are more frequent than negative ones.

As the lexicon's entries are not POS-tagged, we assume that all posible POS-tags of a word have the same polarity. However, this assumption can introduce noise in the training data. For example, the word *ill*, which is labelled as negative by the lexicon, will be labelled as negative for two different POS-tags: adjective, and nominal+verbal contraction. This word is very likely to express a negative sentiment when used as an adjective, but it is unlikely to express a negative sentiment when it refers in a misspelled way to the contraction *I'll*. A simple outlier removal technique to deal with this problem will be discussed in Section 6.2.

Once our time-series are created, we extract from them the word-level features described in Section 5. The feature values obtained for the words used in the time-series visualisation example (Figure 2) are given in Table 5. We can see that each entry has a POS-tag prefix. As expected, location-oriented features from the same time-series exhibit similar values.

A scatterplot between attributes **sgd.mean** and **so.mean** for the labelled words from the STS corpus is shown in Figure 3. From the figure we can observe that the two variables are highly correlated. The correlation is 0.858 and 0.877 for the ED and STS corpora respectively. Positive, negative, and neutral words are depicted with different colours. We can observe that negative words tend to show low values of sgd.mean and so.mean, and positive words tend to show the opposite. Neutral words are more spread out and hard to distinguish. This pattern can also be clearly seen in the boxplots shown in Figure 4. We can observe from the boxplots that the three classes of words can exhibit different statistical properties in both sgd.mean and so.mean. The medians of the classes show an accurate correspondence with the word's polarity, i.e., $\text{median}(pos) > \text{median}(neu) > \text{median}(neg)$. The boxplots also show a substantial number of outliers, especially for neutral words.

Attribute	A-lovely	A-yellow	A-upset	V-worried
sgd.last	0.6	0.3	-2.5	-1.5
sgd.mean	0.9	0.2	-2.4	-1.6
sgd.trunc.mean	0.9	0.2	-2.5	-1.6
sgd.median	0.9	0.3	-2.5	-1.6
sgd.sd	0.3	0.2	0.5	0.5
sgd.sg	0.0	0.0	0.0	0.0
sgd.sg.diff	0.2	0.0	0.1	0.0
sgd.iqr	0.5	0.3	0.3	0.3
so.last	1.5	0.5	-3.7	-2.0
so.mean	1.3	0.4	-3.7	-1.8
so.trunc.mean	1.3	0.4	-3.7	-1.9
so.median	1.3	0.5	-3.7	-1.9
so.sd	0.1	0.2	0.2	0.4
so.sg	0.0	0.0	0.0	0.0
so.sg.diff	0.5	0.4	0.4	0.4
so.iqr	0.1	0.1	0.1	0.1
pos.tag	adjective	adjective	adjective	verb
label	positive	neutral	negative	negative

Table 5: Word-level feature example.

Figure 3: SO vs SGD scatterplot.

Next, we focus on the word-level classification problem. With the aim of gaining a better understanding of the problem, we study three word-level classification problems:

1. *Neutrality*: Classify words as neutral (objective) or non-neutral (subjective). We label positive and negative words as non-neutral for this task.

2. *PosNeg*: Classify words to positive or negative classes. We remove all neutral words for this task.

3. *Polarity*: Classify words to classes positive, negative or neutral. This is the classification problem we aim to solve.

We study the information provided by each feature with respect to the three classification tasks described above. This is done by calculating the information gain of each feature. This score is normally used for decision tree learning and measures the reduction of entropy within each class after performing the best split induced by the feature. The information gain obtained for the different attributes in relation to the three classification tasks is shown in Table 6.

We can observe that variables measuring the location of the SO and SGD time-series tend to be more informative than the ones

Dataset	ED			STS		
Task	Neutrality	PosNeg	Polarity	Neutrality	PosNeg	Polarity
pos-tag	0.062	0.017	0.071	0.068	0.016	0.076
sgd.mean	**0.082**	0.233	0.200	**0.104**	0.276	**0.246**
sgd.trunc.mean	0.079	0.237	0.201	**0.104**	0.276	0.242
sgd.median	0.075	0.233	0.193	0.097	0.275	0.239
sgd.last	0.057	0.177	0.155	0.086	0.258	0.221
sgd.sd	0.020	0.038	0.034	0.030	0.030	0.052
sgd.sg	0.029	0.000	0.030	0.049	0.017	0.062
sgd.sg.diff	0.000	0.000	0.008	0.005	0.000	0.000
sgd.iqr	0.018	0.012	0.019	0.015	0.014	0.017
so.mean	0.079	0.283	**0.219**	0.081	**0.301**	0.232
so.trunc.mean	0.077	**0.284**	0.215	0.079	0.300	0.229
so.median	0.077	0.281	0.215	0.076	0.300	0.228
so.last	0.069	0.279	0.211	0.084	0.300	0.240
so.sd	0.000	0.015	0.008	0.000	0.012	0.007
so.sg	0.013	0.216	0.126	0.019	0.239	0.142
so.sg.diff	0.000	0.012	0.009	0.000	0.000	0.000
so.iqr	0.000	0.000	0.000	0.000	0.008	0.000

Table 6: Information gain values.

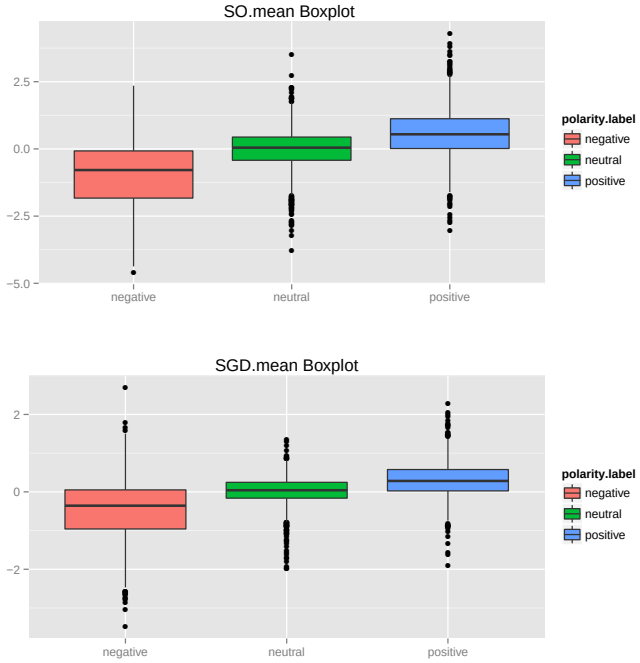

Figure 4: SO and SGD Boxplots.

measuring dispersion. Moreover, the information gain of these variables is much higher for PosNeg than for neutrality. SGD and SO are competitive measures for neutrality, but SO is better for PosNeg. An interesting insight is that features that measure the central tendency of the time-series tend to be more informative than the last values, especially for SGD. These measures smooth the fluctuations of the SGD time-series. We can see that the feature *sgd.mean* is the best attribute for neutrality classification in both datasets. We can also see that POS tags are useful for neutrality detection, but useless for PosNeg. Therefore, we can conclude that positive and negative words have a similar distribution of POS tags.

We trained supervised classifiers for the three different classification problems in both datasets STS and ED. The classification experiments were performed using WEKA[9], a machine learning environment. We studied the following learning algorithms in some preliminary experiments: RBF SVM, Logistic regression, C4.5, and Random Forest. As the RBF SVM produced the best performance among the different methods, we used this method in our classification experiments with a nested grid search procedure for parameter tuning.

The evaluation was done using 10 times 10-folds-cross-validation and different sub-sets of attributes were compared. All the methods are compared with the baseline of using the last value of the semantic orientation, based on a corrected paired t-student test with an alpha level of 0.05 [19]. We used the following sub-sets of attributes: 1) *SO*: Includes only the feature *so.last*. This is the baseline and is equivalent to the standard semantic orientation measure with the decision boundaries provided by the SVM. 2) *ALL*: Includes all the features. 3)*SGD.TS+POS*: Includes all the features from the SGD time-series and the POS tag. 4)*SO.TS+POS*: Includes all the features from the SO time-series and the POS tag. 5)*SO+POS*: Includes the features *so.last* and the POS tag.

We evaluate the classification accuracy and the weighted area under the ROC curves (AUCs) (to deal with class imbalances) for the four different sub-sets of attributes in the two datasets. The classification results are presented in Table 7. The symbols ◦ and • correspond to statistically significant improvements and degradations with respect to the baseline, respectively.

We can observe a much lower performance in Neutrality detection than in PosNeg. This validates the hypothesis that the detection of neutral Twitter words is much harder than distinguishing between positive and negative words. The performance on both datasets tends to be similar. However, the AUC results for STS are better than for ED. This suggests that a collection of balanced positively and negatively labelled tweets could be more suitable for lexicon expansion. Another result is that the combination of all features leads to a significant improvement over the baseline in accuracy and AUC for neutrality and polarity classification. In the PosNeg classification task we can see that the baseline is very strong. This suggests that SO is very good for discriminating between positive and negative words, but not strong enough when neutral words are included. Regarding SO and SGD time-series, we can conclude that they are competitive for neutrality detection. However, SO-based features are better for PosNeg and Polarity tasks.

[9] http://www.cs.waikato.ac.nz/ml/weka/

Accuracy					
Dataset	SO	ALL	SGD.TS+POS	SO.TS+POS	SO+POS
ED-Neutrality	61.52 ± 2.21	**65.16 ± 2.09** ○	64.55 ± 2.27 ○	64.9 ± 2.14 ○	64.18 ± 2.12 ○
ED-PosNeg	74.78 ± 2.93	**76.04 ± 2.72**	73.61 ± 2.51	75.6 ± 2.84	74.99 ± 2.72
ED-Polarity	59.48 ± 2.29	**61.93 ± 2.1** ○	60.97 ± 1.96	61.73 ± 1.98 ○	61.57 ± 2.04 ○
STS-Neutrality	62.99 ± 2.03	**66.2 ± 2.11** ○	65.26 ± 2.34 ○	65.73 ± 1.98 ○	65.77 ± 2.09 ○
STS-PosNeg	**77.18 ± 2.88**	76.98 ± 2.82	75.39 ± 2.89 ●	76.76 ± 2.89	76.98 ± 2.71
STS-Polarity	60.2 ± 2.23	**62.74 ± 1.52** ○	62.34 ± 1.61 ○	62.14 ± 1.73 ○	62.1 ± 1.78 ○
Weighted AUC					
Dataset	SO	ALL	SGD.TS+POS	SO.TS+POS	SO+POS
ED-Neutrality	0.62 ± 0.02	**0.65 ± 0.02** ○	**0.65 ± 0.02** ○	**0.65 ± 0.02** ○	0.64 ± 0.02 ○
ED-PosNeg	0.74 ± 0.03	**0.75 ± 0.03**	0.71 ± 0.03 ●	0.74 ± 0.03	0.73 ± 0.03
ED-Polarity	0.62 ± 0.02	**0.65 ±0.02** ○	0.64 ± 0.02	**0.65 ± 0.02** ○	0.64 ± 0.02 ○
STS-Neutrality	0.63 ± 0.02	**0.67 ± 0.02** ○	0.66 ± 0.02 ○	0.66 ± 0.02 ○	0.66 ± 0.02○
STS-PosNeg	**0.77 ± 0.03**	**0.77 ± 0.03**	0.75 ± 0.03 ●	**0.77 ± 0.03**	**0.77 ± 0.03**
STS-Polarity	0.64 ± 0.02	**0.66 ± 0.01** ○	0.65 ± 0.02 ○	**0.66 ± 0.02** ○	**0.66 ± 0.02** ○

Table 7: World-level classification performance.

6.2 Lexicon expansion

The ultimate goal of the polarity-classification of words is to produce a Twitter-oriented opinion lexicon based on the properties of SentiWordet, i.e., a lexicon of POS-tagged disambiguated entries with their corresponding distribution for positive, negative, and neutral classes. To do this, we fit a logistic regression model to the outputs of the support vector machine trained for the *polarity* problem using all the attributes. The resulting model is then used to classify the remaining unlabelled words. This process is performed for both STS and ED datasets.

A sample from the expanded word list is given in Table 8. We can see that each entry has the following attributes: the word, the POS-tag, the sentiment label that corresponds to the class with maximum probability, and the distribution. We inspected the expanded lexicon and observed that the estimated probabilities tend to agree with the desired values. However, there are some words for which the estimated distribution makes no sense at all, such as the word *same* from the example.

word	POS	label	negative	neutral	positive
alrighty	interjection	positive	0.021	0.087	0.892
boooooo	interjection	negative	0.984	0.013	0.003
lmaoo	interjection	positive	0.19	0.338	0.472
french	adjective	neutral	0.357	0.358	0.285
handsome	adjective	positive	0.007	0.026	0.968
saddest	adjective	negative	0.998	0.002	0
same	adjective	negative	0.604	0.195	0.201
anniversary	common.noun	neutral	0.074	0.586	0.339
tear	common.noun	negative	0.833	0.124	0.044
relaxing	verb	positive	0.064	0.244	0.692
wikipedia	proper.noun	neutral	0.102	0.644	0.254

Table 8: Expanded words example.

The provided probabilities can also be used to explore the sentiment intensities of words. In Figure 5, we visualise the expanded lexicon intensities of words classified as positive and negative through word clouds. The sizes of the words are proportional to the log odds ratios $\log_2(\frac{P(pos)}{P(neg)})$ and $\log_2(\frac{P(neg)}{P(pos)})$ for positive and negative words, respectively.

Next, we study the usefulness of our expanded lexicons ED and STS for Twitter polarity classification. This involves categorising a whole tweet according to positive or negative sentiment classes. Lexicon-based methods have shown to be more effective in distinguishing positive and negative messages than for the detection of subjectivity [4]. Therefore, we omit the detection of neutral tweets. The experiments are performed on three collections of tweets that were manually annotated to positive and negative classes. The first collection is *6HumanCoded*[10], which was used to evaluate SentiStrength [25]. In this dataset tweets are scored according to a positive and negative numerical scores. We use the difference of these scores to create polarity classes and discard messages where this difference is equal to zero. The other datasets are *Sanders*[11], and *SemEval*[12]. The number of positive and negative tweets per corpus is given in Table 9.

We train a logistic regression on the labelled collections of tweets based on simple count-based features extracted using the lexicons. We compute features in the following manner. We count the number of positive and negative words from the metalexicon matching the content of the tweet. From the expanded lexicons we create a positive and a negative score. The positive score is calculated by adding the positive probabilities of POS-tagged words labelled as

(a) (b)

Figure 5: Word clouds of positive and negative words using log odds proportions.

[10] http://sentistrength.wlv.ac.uk/documentation/6humanCodedDataSets.zip

[11] http://www.sananalytics.com/lab/twitter-sentiment/

[12] http://www.cs.york.ac.uk/semeval-2013/task2/

Accuracy				
Dataset	Baseline	ED	STS	Combination
6-coded	71.79 ± 2.79	74.91 ± 2.56 ∘	75.11 ± 2.66 ∘	**75.31 ± 2.42** ∘
Sanders	71.43 ± 3.76	77.17 ± 3.68 ∘	77.32 ± 4.09 ∘	**77.54 ± 3.64** ∘
SemEval	76.81 ± 1.22	76.66 ± 1.38	77.7 ± 1.25 ∘	**78.13 ± 1.38** ∘
Weighted AUC				
Dataset	Baseline	ED	STS	Combination
6-coded	0.77 ± 0.03	0.82 ± 0.03 ∘	0.82 ± 0.02 ∘	**0.83 ± 0.02** ∘
Sanders	0.77 ± 0.04	0.83 ± 0.04 ∘	**0.84 ± 0.04** ∘	**0.84 ± 0.04** ∘
SemEval	0.77 ± 0.02	0.81 ± 0.02 ∘	**0.83 ± 0.02** ∘	**0.83 ± 0.02** ∘

Table 10: Message-level polarity classification performance.

Accuracy				
Dataset	Baseline	ED	STS	Combination
6Coded	71.79 ± 2.79	**76.57 ± 2.3** ∘	75.28 ± 2.5 ∘	76.52 ± 2.43 ∘
Sanders	71.43 ± 3.76	76.78 ± 3.99 ∘	75.83 ± 4.19 ∘	**76.88 ± 4.06** ∘
SemEval	76.81 ± 1.22	78.23 ± 1.3 ∘	77.07 ± 1.18	**78.35 ± 1.24** ∘
Weighted AUC				
Dataset	Baseline	ED	STS	Combination
6Coded	0.77 ± 0.03	**0.84 ± 0.02** ∘	0.82 ± 0.03 ∘	**0.84 ± 0.02** ∘
Sanders	0.77 ± 0.04	**0.83 ± 0.04** ∘	0.82 ± 0.04 ∘	**0.83 ± 0.04** ∘
SemEval	0.77 ± 0.02	0.83 ± 0.02 ∘	0.82 ± 0.02 ∘	**0.84 ± 0.02** ∘

Table 11: Message-level polarity classification performance with outlier removal.

	Positive	Negative	Total
6Coded	1340	949	2289
Sanders	570	654	1224
SemEval	5232	2067	7299

Table 9: Message-level polarity classification datasets.

positive within the tweet's content. The negative score is calculated in an analogous way from the negative probabilities. Words are discarded as non-opinion words whenever the expanded lexicons labelled them as neutral.

We study four different setups based on these attributes. 1) *Baseline*: It includes the attributes calculated from the metalexicon. 2) *ED*: It includes the baseline and the attributes from the ED expanded lexicon. 3) *STS*: This one is analogus to ED, but using the STS lexicon. 4) *Combination*: It includes the baseline, ED, and STS.

In the same way as in the word-level classification task, we rely on the accuracy and the weighted AUC as evaluation measures, and we compare the different setups with the baseline using the corrected paired t-tests. The classification results obtained for the different setups are shown in Table 10.

The results indicate that the expanded lexicons produce meaningful improvements in performance over the baseline on the different datasets. The performance of STS is slightly better than that of ED. This pattern was also observed in the word-level classification performance shown in Table 7. This suggests that the two different ways of evaluating the lexicon expansion, one at the word level and the other at the message level, are consistent with each other. When we rely on the AUC criterion the improvements become more significant on SemEval, which is the largest dataset. We can also observe that the best accuracies are obtained when the attributes from both expanded lexicons are used. Therefore, we can conclude that lexicons expanded from different collections of tweets complement each other to some extent.

As was discussed in the previous section, the metalexicon does not provide POS-tagged entries. Therefore, words exhibiting multiple POS-tags were labelled with the same polarity in the word-level training data. We also mentioned that this assumption could make the classifier learn spurious patterns and erroneously classify unlabelled words. For example the word *ill* together with the POS-tag *nominal+verb* receives a negative label. However, when *ill* is used with this part-of-speech tag that refers to the contraction of the pronoun *I* and the verb *will*, it should be labelled as neutral instead. Moreover, by inspection we realised that *ill* is the only labelled entry with the POS *nominal+verb*. Considering that the POS tag is also used as a feature in the word-level classifiers, we observed that most of the words exhibiting the POS tag *nominal+verb* in both expanded lexicons were classified to the negative class. Common sense suggests that these words should be expanded to the neutral class.

In order to avoid learning this type of spurious pattern, we retrained the word-level classifiers using an outlier removal technique. More specifically, we clean out the instances from the word-level training data that are misclassified by a classifier trained using 10-folds cross-validation. Afterwards, we retrain the word-level classifiers and create new versions of the expanded lexicons for both STS and ED datasets. In the new versions of the expanded lexicons, words exhibiting the *nominal+verb* POS are classified to the neutral class.

The message-level classification results obtained by the expanded lexicons with outlier removal are shown in Table 11. There is im-

provement in performance for the ED lexicon for the *6Coded* and *Sanders* datasets in comparison to the results obtained by the original expanded lexicons. Apparently, the ED corpus is more susceptible to outliers than STS. Additionally, for the datasets *6Coded* and *Sanders* the best accuracies and AUC scores obtained with outlier-cleansed lexicons are better than the best performances obtained with the original lexicons as displayed in Table 10.

7. CONCLUSIONS

In this article, we presented a supervised method for opinion lexicon expansion in the context of Twitter messages. The method creates a lexicon with disambiguated POS entries and a probability distribution for positive, negative, and neutral classes.

To the best of our knowledge, our method is the first approach for creating Twitter opinion lexicons with these characteristics. Considering that these characteristics are very similar to those of SentiWordNet, a well-known publicly available lexical resource, we believe that several sentiment analysis methods that are based on SentiWordnet can be easily adapted to Twitter by relying on our lexicon. The source-code of our method, together with the expanded lexicons are publicly available to the research community[13].

Our supervised framework for lexicon expansion opens several directions for further research. The method could be used to create domain-specific lexicons by relying on tweets collected from the target domain. However, there are many domains such as politics, in which emoticons are not frequently used to express positive and negative opinions. New ways for automatically labelling collections of tweets should be explored. We could rely on other domain-specific expressions such as hashtags, or use message-level classifiers trained from domains in which emoticons are frequently used.

Other types of word-level features based on the context of the word can be explored. We could rely on well-known opinion properties such as negations, opinion shifters, and intensifiers, to create these features.

Another important problem to be studied is how to reduce the noise in the seed lexicon. This could be addressed by manually cleaning the lexicon or by exploring different outlier removal techniques.

We believe that unlabelled words and their feature values could provide valuable information that is not being exploited so far. Semi-supervised methods such as the EM algorithm [21] could be used to include unlabelled words as part of the training process.

As our word-level features are based on time-series, they could be easily calculated in an on-line fashion from a stream of time-evolving tweets. Based on this, we could study the dynamics of opinion words. New opinion words could be discovered because the change of the distribution in certain words could be tracked. This approach could be used for online lexicon expansion in specific domains, and potentially be useful for high impact events on Twitter, such as elections and sports competitions.

8. REFERENCES

[1] *AVAYA: Sentiment Analysis on Twitter with Self-Training and Polarity Lexicon Expansion*, Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013). Association for Computational Linguistics, 2013.

[2] S. Baccianella, A. Esuli, and F. Sebastiani. Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In N. C. C. Chair), K. Choukri, B. Maegaard, J. Mariani, J. Odijk, S. Piperidis, M. Rosner, and D. Tapias, editors, *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta, may 2010. European Language Resources Association (ELRA).

[3] A. Bifet and E. Frank. Sentiment knowledge discovery in twitter streaming data. In *Proceedings of the 13th international conference on Discovery science*, DS'10, pages 1–15, Berlin, Heidelberg, 2010. Springer-Verlag.

[4] F. Bravo-Marquez, M. Mendoza, and B. Poblete. Meta-level sentiment models for big social data analysis. *Knowledge-Based Systems*, 69(0):86 – 99, 2014.

[5] A. Esuli and F. Sebastiani. Determining the semantic orientation of terms through gloss classification. In *Proceedings of the 14th ACM International Conference on Information and Knowledge Management*, CIKM '05, pages 617–624, New York, NY, USA, 2005. ACM.

[6] A. Esuli and F. Sebastiani. Sentiwordnet: A publicly available lexical resource for opinion mining. In *In Proceedings of the 5th Conference on Language Resources and Evaluation (LREC'06*, pages 417–422, 2006.

[7] K. Gimpel, N. Schneider, B. O'Connor, D. Das, D. Mills, J. Eisenstein, M. Heilman, D. Yogatama, J. Flanigan, and N. A. Smith. Part-of-speech tagging for twitter: Annotation, features, and experiments. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2*, pages 42–47. Association for Computational Linguistics, 2011.

[8] A. Go, R. Bhayani, and L. Huang. Twitter sentiment classification using distant supervision. *CS224N Project Report, Stanford*, 2009.

[9] V. Hatzivassiloglou and K. R. McKeown. Predicting the semantic orientation of adjectives. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics and Eighth Conference of the European Chapter of the Association for Computational Linguistics*, ACL '98, pages 174–181, Stroudsburg, PA, USA, 1997. Association for Computational Linguistics.

[10] V. Hatzivassiloglou and J. Wiebe. Effects of adjective orientation and gradability on sentence subjectivity. In *COLING*, pages 299–305, 2000.

[11] M. Hu and B. Liu. Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '04, pages 168–177, New York, NY, USA, 2004. ACM.

[12] L. Jiang, M. Yu, M. Zhou, X. Liu, and T. Zhao. Target-dependent twitter sentiment classification. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 151–160. Association for Computational Linguistics, 2011.

[13] J. Kamps, M. Marx, R. J. Mokken, and M. De Rijke. Using WordNet to Measure Semantic Orientation of Adjectives. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC)*, volume 4, pages 1115–1118. European Language Resources Association (ELRA), May 2004.

[14] S.-M. Kim and E. Hovy. Determining the sentiment of opinions. In *Proceedings of the 20th international conference*

[13] https://github.com/felipebravom/TwitterSentLex

on Computational Linguistics, COLING '04, Stroudsburg, PA, USA, 2004. Association for Computational Linguistics.

[15] E. Kouloumpis, T. Wilson, and J. Moore. Twitter sentiment analysis: The good the bad and the omg! *ICWSM*, 11:538–541, 2011.

[16] B. Liu. *Sentiment Analysis and Opinion Mining*. Synthesis Lectures on Human Language Technologies. Morgan & Claypool Publishers, 2012.

[17] S. Mohammad and P. D. Turney. Crowdsourcing a word-emotion association lexicon. *Computational Intelligence*, 29(3):436–465, 2013.

[18] S. M. Mohammad, S. Kiritchenko, and X. Zhu. Nrc-canada: Building the state-of-the-art in sentiment analysis of tweets. *Proceedings of the seventh international workshop on Semantic Evaluation Exercises (SemEval-2013)*, 2013.

[19] C. Nadeau and Y. Bengio. Inference for the generalization error. *Machine Learning*, 52(3):239–281, 2003.

[20] F. r. Nielsen. A new ANEW: Evaluation of a word list for sentiment analysis in microblogs. In *ESWC2011 Workshop on Making Sense of Microposts*, May 2011.

[21] K. Nigam, A. McCallum, and T. Mitchell. Semi-supervised text classification using em. *Semi-Supervised Learning*, pages 33–56, 2006.

[22] A. Pak and P. Paroubek. Twitter as a corpus for sentiment analysis and opinion mining. *Proceedings of LREC 2010*, 2010.

[23] S. Petrović, M. Osborne, and V. Lavrenko. The edinburgh twitter corpus. In *Proceedings of the NAACL HLT 2010 Workshop on Computational Linguistics in a World of Social Media*, WSA '10, pages 25–26, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[24] P. J. Stone, D. C. Dunphy, M. S. Smith, and D. M. Ogilvie. *The General Inquirer: A Computer Approach to Content Analysis*. MIT Press, Cambridge, MA, 1966.

[25] M. Thelwall, K. Buckley, and G. Paltoglou. Sentiment strength detection for the social web. *JASIST*, 63(1):163–173, 2012.

[26] P. D. Turney. Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 417–424, Stroudsburg, PA, USA, 2002. Association for Computational Linguistics.

[27] P. D. Turney and M. L. Littman. Measuring praise and criticism: Inference of semantic orientation from association. *ACM Transactions on Information Systems (TOIS)*, 21(4):315–346, 2003.

[28] Y. Wilks and M. Stevenson. The grammar of sense: Using part-of-speech tags as a first step in semantic disambiguation. *Natural Language Engineering*, 4(02):135–143, 1998.

[29] T. Wilson, J. Wiebe, and P. Hoffmann. Recognizing Contextual Polarity in Phrase-Level Sentiment Analysis. In *Proceedings of Human Language Technologies Conference/Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP 2005)*, Vancouver, CA, 2005.

[30] T. Zhang. Solving large scale linear prediction problems using stochastic gradient descent algorithms. In *Proceedings of the Twenty-first International Conference on Machine Learning*, ICML '04, pages 116–, New York, NY, USA, 2004. ACM.

[31] Z. Zhou, X. Zhang, and M. Sanderson. Sentiment analysis on twitter through topic-based lexicon expansion. In H. Wang and M. Sharaf, editors, *Databases Theory and Applications*, volume 8506 of *Lecture Notes in Computer Science*, pages 98–109. Springer International Publishing, 2014.

[32] C. Zirn, M. Niepert, H. Stuckenschmidt, and M. Strube. Fine-grained sentiment analysis with structural features. In *Proceedings of the 5th International Joint Conference on Natural Language Processing (IJCNLP)*, pages 336–344, 2011.