ECAI 2016
G.A. Kaminka et al. (Eds.)

doi:10.3233/978-1-61499-672-9-498

Annotate-Sample-Average (ASA): A New Distant Supervision Approach for Twitter Sentiment Analysis

Felipe Bravo-Marquez and **Eibe Frank** and **Bernhard Pfahringer** [1]

Abstract. The classification of tweets into polarity classes is a popular task in sentiment analysis. State-of-the-art solutions to this problem are based on supervised machine learning models trained from manually annotated examples. A drawback of these approaches is the high cost involved in data annotation. Two freely available resources that can be exploited to solve the problem are: 1) large amounts of unlabelled tweets obtained from the Twitter API and 2) prior lexical knowledge in the form of opinion lexicons. In this paper, we propose Annotate-Sample-Average (ASA), a distant supervision method that uses these two resources to generate synthetic training data for Twitter polarity classification. Positive and negative training instances are generated by sampling and averaging unlabelled tweets containing words with the corresponding polarity. Polarity of words is determined from a given polarity lexicon. Our experimental results show that the training data generated by ASA (after tuning its parameters) produces a classifier that performs significantly better than a classifier trained from tweets annotated with emoticons and a classifier trained, without any sampling and averaging, from tweets annotated according to the polarity of their words.

1 Introduction

Twitter[2] is a service in which users can post messages or tweets limited to 140 characters and subscribe to tweets posted by other users. It has become the most popular microblogging platform, with hundreds of millions of users who produce millions of posts on a daily basis. The great volume of publicly available social data that is published in Twitter has made it a rich resource for sentiment analysis [9].

A popular approach for classifying tweets (posts in Twitter) into polarity classes is to represent tweets from a corpus of hand-annotated tweets by feature vectors and train supervised models on them [18]. However, considering that annotation of tweets into sentiment classes is a time-consuming and labour-intensive task, supervised models can be impractical in the absence of labelled tweets.

Distant supervision models are heuristic labelling functions [16] for creating training data from unlabelled corpora. These models have been widely adopted for Twitter sentiment analysis because large amounts of unlabelled tweets can be easily obtained through the use of the Twitter API. A well-known distant supervision approach for Twitter polarity classification is the emoticon-annotation approach (EAA), in which tweets with positive :) or negative :(emoticons are labelled according to the polarity indicated by the emoticon after removing the emoticon from the content [22]. This method is affected by two main limitations:

1. The removal of all tweets without emoticons may cause a loss of valuable information.
2. There are many domains such as politics, in which emoticons are not frequently used to express positive and negative opinions.

Opinion lexicons are another type of resource that has been used for supporting the sentiment analysis of tweets. An opinion lexicon is a list of terms or *opinion words* annotated according to sentiment categories such as positive and negative. Examples of positive opinion words are **love** and **happy**, and examples of negative opinion words are **disgusting** and **horrible**. There are several opinion lexicons freely available on the web, e.g., *SentiWordNet*[3], *MPQA Subjectivity Lexicon*[4], and *AFINN*[5]. Opinion lexicons can be used as prior lexical knowledge for calculating the sentiment of tweets [27], and to extract message-level features for training classifiers [3, 9, 18].

In this paper we propose a distant supervision method called **Annotate-Sample-Average** (ASA) for training polarity classifiers in Twitter in the absence of labelled data. ASA takes a collection of unlabelled tweets and a polarity lexicon composed of positive and negative words and creates synthetic labelled instances for Twitter polarity classification. Each labelled instance is created by sampling with replacement a number of tweets containing at least one word from the lexicon with the desired polarity, and averaging the feature vectors of the sampled tweets. This allows the usage of any kind of features for representing the tweets, e.g., unigrams and part-of-speech tags (POS) tags.

Polarity lexicons are normally formed by thousands of opinion words, so there is a high probability that a tweet contains at least one word from the lexicon, which means that ASA can potentially exploit more unlabelled data than EAA because the latter is based on a small number of positive and negative emoticons.

The intuition behind ASA is that a tweet containing a word with a certain known positive or negative polarity has a certain likelihood of expressing the same polarity in the whole message. Of course, the opposite polarity may also be expressed due to the presence of negation, sarcasm, or other opinion words with the opposite polarity. We propose a hypothesis, which we refer to as the "lexical polarity hypothesis", stating that the first scenario is more likely than the second one. Based on that, when sampling and averaging multiple tweets exhibiting at least one word with the desired positive or negative polarity, we increase the confidence of obtaining a vector located

[1] University of Waikato, New Zealand, email: fjb11@students.waikato.ac.nz, {eibe,bernhard}@cs.waikato.ac.nz
[2] http://www.twitter.com
[3] http://sentiwordnet.isti.cnr.it/
[4] http://mpqa.cs.pitt.edu/opinionfinder/opinionfinder_2/
[5] http://neuro.imm.dtu.dk/wiki/AFINN

in the region of the desired polarity.

Most sentiment analysis datasets are imbalanced in favor of positive examples [12]. This is presumably because users are more likely to report positive than negative opinions. The shortcoming of training sentiment classifiers from imbalanced datasets is that many classification algorithms tend to predict test samples as the majority class [7] when trained from this type of data. A popular way to address this problem is to rebalance the data by under-sampling the majority class or by over-sampling the minority class. A noteworthy property of ASA is that it incorporates a rebalancing mechanism in which balanced training data can be generated.

We compare classifiers trained with ASA against other distant supervision methods on three collections of hand-annotated polarity tweets. The baselines we consider are EAA and a lexicon-based annotation approach (LAA) that annotates tweets according to the polarity of their words. The experimental results show that ASA, with appropriate choice of the number of tweets averaged for each generated instance, outperforms the other methods in all cases.

This article is organised as follows. In Section 2, we provide a review of related work. In Section 3, we describe the proposed ASA method. The lexical polarity hypothesis is empirically studied in Section 4. The evaluation of the method is presented in Section 5. The conclusions are discussed in Section 6.

2 Related Work

State-of-the art solutions for Twitter polarity classification are based on supervised techniques such as logistic regression and support vector machines trained from hand-annotated polarity corpora. Some of the features used for describing the tweets are: n-grams, POS tags, Brown clusters [4], and features derived from polarity lexicons [18].

In the absence of training data most previous distant supervision approaches for Twitter sentiment analysis rely on strong sentiment signals such as emoticons or hashtags e.g., #joy, #sadness, for labelling tweets into positive and negative polarity classes, after dropping these signals from the content [6].

Emoticon-annotated tweets have been used for a variety of sentiment analysis tasks: training of polarity classifiers [6, 20], training incremental classifiers from Twitter streams [2], fitting sentiment-oriented language models [14], inducing polarity lexicons [18], and initialising the parameters of deep neural networks [23, 26].

Other types of knowledge are lexical knowledge provided by opinion lexicons and contextual knowledge provided by unlabelled corpora. There is some work exploiting these sources of knowledge for training document-level sentiment classifiers from small collections of labelled documents. In [24], words and documents are jointly represented by a bipartite graph of labelled and unlabelled nodes. The sentiment labels of words and documents are propagated to the unlabelled nodes using regularised least squares. In [13], the term-document matrix associated with a corpus of documents is factorised into three matrices specifying cluster labels for words and documents using a constrained non-negative tri-factorisation technique. Sentiment-annotated words and documents are introduced into the model as optimisation constraints. A generative naive Bayes model based on a polarity lexicon, which is then refined using sentiment-annotated documents, is proposed in [15]. Zhang et al. [29] proposed a lexicon-based approach for annotating unlabelled tweets into polarity classes regarding a given entity by aggregating the polarities of words from a lexicon with positive and negative words using a scoring function. The automatically labelled tweets are then used for training a classifier.

Another approach based on distant supervision and lexical prior knowledge is proposed in [25]. The authors build a graph that has users, tweets, words, hashtags, and emoticons as its nodes. A subset of these nodes is labelled by prior sentiment knowledge provided by a polarity lexicon, the known polarity of emoticons, and a message-level classifier trained with emoticons. These sentiment labels are propagated throughout the graph using random walks.

A semi-supervised model for imbalanced sentiment classification is proposed in [12]. The model exploits both labelled and unlabelled documents by iteratively performing under-sampling of the majority class in a co-training framework using random subspaces of features.

The ASA method proposed in this paper exploits prior lexical knowledge and unlabelled data for creating synthetic polarity data by sampling and averaging multiple tweets without requiring any labelled tweets. ASA works on the whole message rather than being entity oriented as the method in [29]. Moreover, ASA can be used for creating training data with any size and distribution of labels and hence may be useful for dealing with the class imbalance problem reported in [12]. To the best of our knowledge, this is the first distant supervision method for Twitter sentiment analysis with these characteristics.

3 Annotate-Sample-Average Method

In this section, we describe the Annotate-Sample-Average (ASA) algorithm for generating training data for Twitter polarity classification. The method receives two data inputs: 1) the source corpus, and 2) the opinion lexicon.

The source corpus is a collection of unlabelled tweets $\mathcal{C}$ on which the generated instances are based. The corpus can be built using the public Twitter API[6], which allows the retrieval of public tweets. The tweets must be written in the same language as the opinion lexicon, and the type of tweets included in the collection should depend on the type of sentiment classifier intended to be built. For instance, in order to build a domain-specific sentiment classifier (e.g., for a political election), the collection should be restricted to tweets associated with the target domain. This can be done using the Twitter API by specifying key words, users, or geographical areas. In this work, we focus on domain-independent polarity classification. Thus, we consider a general purpose collection of English tweets.

The opinion lexicon $\mathcal{L}$ is a list of words labelled by sentiment. In this work, we consider positive and negative sentiment categories. The positive and negative subsets of the lexicon are denoted by symbols $\mathcal{L}_+$ and $\mathcal{L}_-$ respectively. Several existing opinion lexicons can be used here. There are basically two families of lexicons that can be considered:

1. Manually annotated lexicons, in which the sentiment of the words is annotated according to human judgements. Crowdsourcing tools such as Amazon Mechanical Turk can be used to support the annotation [17].
2. Automatically-annotated lexicons that are created by automatically expanding a small set of opinion words using relations provided by semantic networks, e.g., synonyms, and antonyms [8], or using statistical associations calculated from document corpora, e.g., point-wise mutual information [28].

Manually-annotated lexicons tend to be smaller than the automatically made ones. Conversely, automatically-annotated lexicons are likely to be noisy and may include several neutral words that are not

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

very useful for polarity classification [3]. In this work, we use the AFINN lexicon [1], which is a manually annotated lexicon formed by 1176 positive words and 2204 negative words. The lexicon includes informal words commonly found in Twitter such as slang, obscene words, acronyms and Web jargon. It is important to mention that AFINN does not include any emoticons.

The other parameters of ASA are: a, which determines the number of tweets to be averaged for each generated instance, p, which corresponds to the number of positive instances to be generated, n, corresponding to the number of negative instances, and m, which is a flag specifying how to handle tweets with both positive and negative words.

The tweets from $\mathcal{C}$ are preprocessed according to the procedure proposed in [6]. All tweets are lowercased and tokenised. The words are simplified by replacing sequences of letters occurring more than two times with two occurrences of the letter (e.g., huuungry is reduced to huungry, loooove to loove) and replacing user mentions and URLs with the generic tokens "USER" and "URL", respectively.

The first step of the algorithm is the **annotation** phase, in which the tweets from $\mathcal{C}$ are annotated according to the prior sentiment knowledge provided by the lexicon. Every time a positive word from $\mathcal{L}_+$ is found in a message, the whole tweet is added to a set called **posT**; analogously, if a negative word is found in $\mathcal{L}_-$, the tweet is added to a set called **negT**. Tweets with both positive and negative words will be discarded if the flag m is set, and will be simultaneously added to both **posT** and **negT** otherwise.

The tweets contained in **posT** and **negT** are candidates for building the synthetic labelled instances. The assumption here is that tweets in each set, positive and negative, are more probable to express the corresponding polarity in the whole message than the opposite polarity. This can be explained by the the short length of tweets. As tweets are short straight-to-the-point messages, the presence of a polarity word has a strong correlation with the overall polarity expressed in the message. For example, the tweet: *"Hey guess what? I think you're awesome"* contains the word *awesome* and is clearly expressing a positive sentiment. Conversely, there also tweets with opinion words than can express the opposite polarity, e.g., *"Not happy where I'm at in life"*. This can occur due to several factors such as the presence of other words with the opposite polarity, negations, or sarcasm. However, we hypothesise that the first situation is more likely than the the second one. We refer to this hypothesis as the "lexical polarity hypothesis" and we study it empirically in Section 4.

We represent all the candidate tweets by vectors of features. We consider three type of features, which are concatenated for building the feature space. These features have been proven to be useful for analysing the sentiment of tweets [18]:

1. Word unigrams (UNI): a vector space model based on counting the frequency of unigrams.
2. Brown clusters (BWN): a vector space model based on counting the frequency of word clusters trained with the Brown clustering algorithm [4]. This algorithm produces hierarchical clusters of words by maximising the mutual information of bigrams.
3. Part-of-speech tags (POS): a vector space model based on counting the frequency of each POS tag in the message.

The second step of ASA is the **sampling** step. ASA randomly samples with replacement a tweets from either **posT** or **negT** for each generated instance. Next, in the **averaging** step the feature vectors of the sampled tweets are averaged and labelled according to the polarity of the set from which they were sampled. The rationale behind this step is that, assuming that the "lexical polarity hypothesis" holds, averaging multiple tweets sampled from the same set increases the confidence of generating instances located in the region of the desired polarity.

We define the random variable D as the event of sampling a tweet with the desired positive or negative polarity from either **posT** or **negT**. We assume that D is distributed with a Bernoulli distribution of parameter p_d. According to the lexical polarity hypothesis, $p_d > 0.5$. We define another random variable M as the event that the majority of the a randomly sampled tweets from **posT** or **posN** have the desired polarity. This is equivalent to saying that at least $\lfloor \frac{a}{2} \rfloor + 1$ tweets from the sample have the desired positive or negative polarity. If we assume that the tweets in **posT** and **negT** are independent and identically distributed (IID), the probability of M can be calculated by adding the values of the Binomial probability mass function from $\lfloor \frac{a}{2} \rfloor + 1$ to a. This corresponds to adding all the cases in which more than the half of the sampled tweets (the majority) have the desired polarity. This probability is calculated as follows:

$$P(M) = \sum_{i=\lfloor \frac{a}{2} \rfloor + 1}^{a} \binom{a}{i} p_d^i (1-p_d)^{a-i}$$

Note that this value is equivalent to 1 minus the cumulative distribution function of the Binomial distribution evaluated at $\lfloor \frac{a}{2} \rfloor$. The probabilities of M for different values of a ($a \geq 3$) and p_d ($p_d > 0.5$) are shown in Table 1.

	$p_d = 0.6$	$p_d = 0.7$	$p_d = 0.8$	$p_d = 0.9$
$a = 3$	0.648	0.784	0.896	0.972
$a = 5$	0.683	0.837	0.942	0.991
$a = 10$	0.633	0.850	0.967	0.998
$a = 50$	0.902	0.998	1	1
$a = 100$	0.973	1	1	1
$a = 500$	1	1	1	1
$a = 1000$	1	1	1	1

Table 1. Probabilities of sampling a majority of tweets with the desired polarity.

From the table, we observe that all the calculated probabilities are greater than p_d and generally increase when increasing p_d or a (exceptions occur when switching from an odd to an even number of votes). Thus, assuming the lexical polarity hypothesis is true and $p_d > 0.5$ for **posT** and **negT**, we can say that the majority of the tweets sampled by ASA have the desired polarity with a probability greater than p_d. We can expect that the instances produced by ASA will behave similarly to the majority of the instances they are obtained from. Thus, compared to sampling individual tweets, we can have greater confidence that ASA instances will be in the desired polarity region.

The ideas discussed above are inspired by Condorcet's Jury Theorem, which is used in the context of decision making. The theorem states that if a random individual votes for the correct decision with probability $p_d > 0.5$, the probability of the majority being correct tends to one when increasing the number of independent voters. This is a consequence of the law of great numbers, and as was shown in [10], the same conclusions can be obtained after relaxing the independence assumption.

In our problem, each tweet sampled from **posT** or **negT** can be interpreted as a vote for the polarity of the averaged instance. We expect a trade-off in the value of a. While a small value of a will decrease the confidence of generating an instance with the target

Algorithm ASA ($\mathcal{C}$, $\mathcal{L}$, a, p,n,m)
  foreach *tweet* $\in \mathcal{C}$ **do**
    if m *and (hasWord(tweet,*$\mathcal{L}_+$*) and hasWord(tweet,*$\mathcal{L}_-$*))* **then**
      continue
    if *hasWord(tweet,*$\mathcal{L}_+$*)* **then**
      tweetVec ← extractFeatures(tweet)
      posT.put(tweetVec)
    if *hasWord(tweet,*$\mathcal{L}_-$*)* **then**
      tweetVec ← extractFeatures(tweet)
      posN.put(tweetVec)
  end
  $i \leftarrow 0$
  while $i \leq p$ **do**
    pInst ← sampleAndAverage(*posT,a*)
    pInst.label ← pos
    $\mathcal{O}$.put(pInst)
    $i \leftarrow i + 1$
  end
  $i \leftarrow 0$
  while $i \leq n$ **do**
    nInst ← sampleAndAverage(*negT,a*)
    nInst.label ← neg
    $\mathcal{O}$.put(nInst)
    $i \leftarrow i + 1$
  end
  return $\mathcal{O}$;
Procedure sampleAndAverage(*T,a*)
  $i \leftarrow 0$
  inst ← newZeroVector
  while $i \leq a$ **do**
    x ← randomSample(T)
    inst ← inst + (x/a)
    $i \leftarrow i + 1$
  end
  return inst;

Algorithm 1: ASA ALGORITHM

polarity, a very large value will generate instances that, despite being likely to have the right label, will be very similar to each other. This could affect the generalisation ability of a classifier trained from those instances.

The resulting training dataset $\mathcal{O}$ is created by repeating the sample and average steps p times for the positive class and n times for the negative one. The pseudo-code of ASA is given in Algorithm 1.

Setting the flag m in the algorithm will generate polarity instances from tweets in which words from the opposite polarity are never observed. Considering that positive and negative tweets are likely to contain words with the opposite polarity, we expect that unsetting the flag will produce instances with better generalisation properties. Both setups are compared in Section 5.

We use ASA for creating balanced training data by setting p and n to the same value. This is done to address the sentiment imbalance problem discussed in [12]: classifiers trained from imbalanced datasets may have difficulties recognising the minority class. The balancing properties of ASA are inspired by a well-known resampling technique used for training classifiers from imbalanced datasets called Synthetic Minority Over-sampling Technique (SMOTE) [5]. SMOTE oversamples the minority class by generating synthetic examples for the minority class. Each new instance is calculated as a random weighted average between an existing example of the minority class and one of its nearest neighbours. The similarity between ASA and SMOTE is that both methods generate new instances by averaging existing ones. The difference is that in ASA the average is unweighted and can involve more than two examples. Furthermore, ASA does not require calculating the distance between the examples being averaged. This is a convenient aspect of ASA considering that tweets are represented by high-dimensional vectors. Another important difference relates to the type of data used for generating the instances. SMOTE combines labelled instances; ASA combines unlabelled instances annotated using an opinion lexicon.

4 The Lexical Polarity Hypothesis

In this section, we study the lexical polarity hypothesis on which ASA is based. It encapsulates the idea that a single opinion word in a tweet is a very strong indicator of the polarity of the message. The hypothesis is expressed in the following two statements:

1. A tweet containing at least one positive word is more likely to be positive than negative.
2. A tweet containing at least one negative word is more likely to be negative than positive.

We study this hypothesis empirically by estimating the probabilities of events corresponding to these statements using the *SemEval*[7] corpus of hand-annotated positive and negative tweets and the AFINN lexicon. The *SemEval* [19] corpus contains 5232 positive tweets and 2067 negative tweets, annotated by human evaluators using the crowdsourcing platform Amazon Mechanical Turk[8]. Each tweet is annotated by five Mechanical Turk workers and the final label is determined based on the majority of the labels. We take a balanced sample of 2000 positive and 2000 negative tweets from this corpus to avoid bias caused by unevenly distributed tweets and focus the analysis on how the polarity of tweets is affected by the polarity of their words. Hence, we calculate the sets **posT** and **negT** from this corpus and study the polarity distribution of their messages.

We first study the distribution of **posT** and **negT** by unsetting the m flag. Hence, we include tweets with mixed positive and negative words in both sets. The set **postT** has 2419 tweets, which corresponds to 60% of the tweets, and has a distribution of 826 negative and 1593 positive tweets. Thus, the estimated probability of a tweet from **posT** of having a positive polarity is 0.66. The set **negT** contains 1774 tweets, corresponding to 44% of the tweets, and has a distribution of 1354 negative and 420 positive tweets. This gives an estimated probability of 0.76 that a tweet from **negT** is negative. These results suggest that negative words are stronger indicators than positive words for determining the polarity of a tweet.

We also study the distribution of **posT** and **negT** after discarding tweets with mixed positive and negative words (m turned on). In this case, the size of **posT** is reduced to 1552 (39% of the total) tweets with a distribution of 284 negative and 1268 positive tweets. This gives an estimated probability of 0.817 that a tweet from **posT** is positive. The size of **negT** is reduced to 907 tweets (23% of the total) with a distribution of 812 negative and 95 positive tweets. This gives an estimated probability of 0.9 that a tweet from **negT** is negative.

The polarity distribution of these sets is presented as bar charts in Figure 1. The figure shows how the distributions become more skewed when removing tweets with mixed positive and negative opinion words.

[7] http://www.cs.york.ac.uk/semeval-2013/task2/
[8] http://www.mturk.com

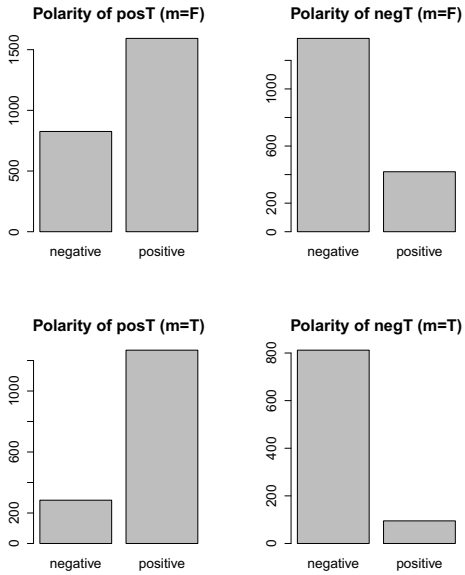

Figure 1. Polarity distributions of **posT** and **negT**.

We also study the distribution of tweets with mixed positive and negative words. We found 857 such tweets (21% of the total) with a distribution of 542 negative and 325 positive tweets. These numbers also indicate that negative opinion words have a greater effect than positive words on the polarity of the tweets in which they occur. However, negative words are also less frequent than positive ones.

The results obtained in this section support the lexical polarity hypothesis on which ASA is based. We can conclude that opinion words are indeed strong indicators of the polarity of tweets. We observed that discarding tweets with mixed opinion words produces a stronger effect. However, it is important to bear in mind that discarding these tweets may also cause loss of valuable information. The further effects of averaging multiple tweets containing opinion words with the same polarity are investigated in the following section.

5 Classification Experiments

In this section, we conduct an experimental evaluation of the proposed ASA algorithm. We evaluate the performance of classifiers trained with instances generated by ASA on different datasets of hand-annotated tweets. We compare these classifiers with classifiers trained on data obtained from two other distant supervision baselines:

1. Emoticon-annotation approach (EAA): labels tweets with positive or negative emoticons according to the emoticon's polarity after removing the emoticon from the message. Tweets containing both positive and negative emoticons are discarded.
2. Lexicon-annotation approach (LAA): annotates tweets using the AFINN lexicon. Tweets with at least one positive word and no negative words are labelled positive, and analogously, tweets with at least one negative word and no positive words are labelled negative. The positive and negative instances from LAA are equivalent to the sets **posT** and **negT** from ASA when m is turned on.

The goal of comparing ASA against EAA and LAA is to determine which distant supervision model generates better training data for polarity classification from a source corpus of unlabelled tweets. Considering that positive signals such as positive emoticons or positive opinion words occur more frequently in tweets than their negative counterparts, we also study balanced versions of EAA and LAA. The balanced baselines are referred to as EAA_B and LAA_B, and are obtained by undersampling the majority class in each case.

The unlabelled tweets used as source corpora in all methods are taken from the Edinburgh corpus (ED) [21], which is a general purpose collection of 97 million unlabelled tweets in multiple languages collected with the Twitter streaming API between November 11th 2009 and February 1st 2010. Tweets written in languages different from English are discarded, resulting in a corpus of around 50 million English tweets.

The positive and negative emoticons used for labelling tweets with EAA are: ":)", ":D", "=D", "=)", ":]", "=]", ":-)", ":-D", ":-]", ";)", ";D", ";]", ";-)", ";-D", and ";-]" for positive tweets and ":(", "=(", ";(", ":[", "=[", ":-(", ":-[", ":(", ":[", and "D:", for negative tweets.

The features used for representing the tweets in both approaches are: unigrams, POS tags, and Brown word clusters. The tokenisation and the POS tagging of the tweets is conducted using the **TweetNLP** library[9]. The word clusters are also taken from the **TweetNLP** project and correspond to 1000 different clusters trained from a collection of around 56 million tweets using the Brown-clustering algorithm.

With the aim of analysing the effect of averaging multiple tweets for building training instances, we study different values of the a parameter of ASA. We also study the effect of including or excluding tweets with mixed positive and negative words by comparing the performance of ASA with the flag m turned on and off, respectively. We create balanced training datasets with size equal to 1% of the size of the source corpus by setting parameters p and n to 0.5% of the source corpus size.

The evaluation of ASA, EAA, and LAA is carried out by comparing the performance obtained by an L_2-regularised logistic regression trained with data generated from those two models, applied on three test collections of tweets that were manually assigned to positive and negative classes. These collections are: *6HumanCoded*[10], *Sanders*[11], and *SemEval*[12], which was also used in Section 4. The *6HumanCoded* dataset is a collection of tweets scored according to positive and negative numerical scores by six human evaluators. The ratings are averaged and we use the difference of these scores to create polarity classes. We discard messages where this difference is zero. The *Sanders* dataset consists of 570 positive and 654 negative tweets evaluated by a single human annotator. The number of positive and negative tweets per dataset is given in Table 2.

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

Table 2. Manually-annotated collections of tweets.

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

The tweets from the target collections are mapped into the same feature-space as the tweets generated by the distant supervision models. The logistic regression model is taken from LIBLINEAR[13], with the regularisation parameter C set to 1.0. Each distant supervision model is trained ten times on data generated from ten independent partitions of 2 million tweets from the source corpus. The average performance of each classifier trained with ASA is compared with the average performance of classifiers trained with each of the four distant supervision baselines 1) EAA, 2) EAA_B, 3) LAA, 4) LAA_B, using a paired Wilcoxon signed-rank test with the significance value set to 0.05.

Different distant supervision models produce different numbers of labelled instances from the same corpus of unlabelled tweets. The average number of positive and negative instances generated by each distant supervision model from the ten collections of 2 million unlabelled tweets is shown in Table 3.

	Avg. Positive	(%)	Avg. Negative	(%)	Avg. Total	(%)
EAA	130,641	(6.5%)	21,537	(1.1%)	152,179	(7.6%)
EAA_B	21,537	(1.1%)	21,537	(1.1%)	43,074	(2.2%)
LAA	681,531	(34.1%)	294,177	(14.7%)	975,708	(48.8%)
LAA_B	294,177	(14.7%)	294,177	(14.7%)	588,354	(29.4%)
ASA	10,000	(0.5%)	10,000	(0.5%)	20,000	(1%)

Table 3. Average number of positive and negative instances generated by different distant supervision models from 10 collections of 2 million tweets.

We use the macro-averaged F1 score and the weighted area under the ROC curves (AUCs) as evaluation measures for comparing classifiers. Macro-averaged F1 was used in the SemEval sentiment analysis in Twitter task[14], and AUC is a useful metric for comparing the performance of classifiers because it is independent of any specific value for the decision threshold.

The comparisons are done for each target collection of tweets and the results for the macro-averaged F1 score and AUC are given in Table 4. The statistical significance tests of each configuration of ASA with respect to each of the four baselines are indicated by a sequence of four symbols. Improvements are denoted by a plus (+), degradations by a minus (-), and cases where no statistical significant difference is observed by an equals (=). The baselines are also compared amongst each other.

We observe that EAA performs substantially worse than the other baselines in F1 score. EAA_B performs substantially better than EAA. From Table 3 we observe that EAA is the model that produces the most uneven distribution of positive and negative instance. This suggest that the macro-average F1 score is very sensitive to classifiers trained from heavily imbalanced data. In contrast, we can note that balancing EAA does not cause any improvement in AUC. AUC is a more robust measure for classifiers trained from imbalanced datasets.

Regarding the LAA baseline, we observe a degradation in F1 after balancing the data (LAA_B). On the other hand, LAA_B performs almost identically to LAA in AUC. We believe that the reason why balancing is not causing a positive impact in the lexicon-based approach is that LAA produces a less skewed distribution of positive and negative instances than EAA. The benefits of resampling are more substantial for F1 for very skewed distributions such as those produced by EAA. There is no clear consensus about which baseline is the best. The baselines based on lexicons perform better than the ones based on emoticons in SemEval for both F1 and AUC. In Sanders, the lexicon and the balanced emoticons behave similarly in F1, but the emoticons perform better for AUC. In 6HumanCoded EAA_B performs better than LAA and LAA_B in F1, but in AUC they produce almost identical results. It is worth mentioning that the emoticon-based approach can achieve competitive results to the lexicon-based one even though it generates substantially less training data (Table 3).

Regarding ASA, we observe that the performance achieved by our proposed method depends on the parameter setting. When the tweets with mixed positive and negative tweets are discarded (m=T) we observe that the best results are achieved when very few tweets are averaged. There is a strong decline in the performance of ASA (m=T) when the value of a is increased. We believe that this is because instances become too similar when formed by averaging too many tweets. ASA (m=T) with a=1 is essentially a subsampled version of LAA_B, and indeed produces very similar results. ASA (m=T) is not capable of producing statistically significant improvements over the four baselines for either AUC and F1 score for any dataset, even with its optimum value of a. This suggests that there is no clear contribution in the sample and average steps of ASA when tweets with mixed positive and negative tweets are discarded.

On the oher hand, when tweets with mixed positive and negative words are simultaneously added to both sets (m=F), ASA produces statistically significant improvements over all the baselines in all target collections for both F1 and AUC, for appropriate values of a. The best value of a is ten in the three target collections, for both performance metrics. These results indicate that ASA, with calibrated parameters, outperforms existing distant supervision models for Twitter polarity classification. The fact that turning m off is better than discarding tweets with mixed positive and negative words suggests that mixed tweets contribute to better generalisation. This is because real positive and negative tweets are likely to contain words with both polarities.

We clearly observe that setting a to one in ASA (m=F) produces results that are far from the optimum. This validates the idea that averaging multiple tweets with at least one word with the same polarity increases the chance of producing an instance of the desired polarity. We observe again a decline in performance when the value of a is further increased.

Based on the numbers in Table 3 we are using 7.6 and 48.8 times more training data with EAA and LAA than with ASA respectively. It is noteworthy that ASA classifiers outperform the classifiers trained with EAA and LAA even though they are trained with less data. This essentially shows that ASA can produce a more compact and efficient training dataset than previous distant supervision models.

Examples of tweets from the SemEval corpus classified using ASA, with $a = 10$ and $m = F$, are given in Table 5. The positive and negative words from the AFINN lexicon are marked with blue and red colours respectively. The classification outputs reveal some insights about the strengths and shortcomings of our method. The correctly classified examples suggest that ASA is capable of learning sentiment expressions that go beyond the lexicon used in the annotation phase. This is observed in the second and third negative examples, and the last positive one, which are all correctly classified even though they do not contain AFINN words with the same polarity than the corresponding tweet. ASA learns opinion words co-occurring with the words from the lexicon, because all words from a tweet are considered in the feature space. This is an indirect form of polarity lexicon expansion. Regarding the misclassified examples, we observe that the current implementation of ASA is not capable of accurately handling complex sentiment patterns involving negations

[13] http://www.csie.ntu.edu.tw/~cjlin/liblinear/
[14] http://alt.qcri.org/semeval2016/task4/

Macro-averaged F1			
	6HumanCoded	Sanders	SemEval
EAA_U	0.576 ± 0.007 = - - -	0.506 ± 0.018 = - - -	0.591 ± 0.018 = - - -
EAA_B	0.735 ± 0.008 + = + +	0.709 ± 0.018 + = = =	0.711 ± 0.006 + = - =
LAA_U	0.729 ± 0.004 + - = +	0.711 ± 0.003 + = = +	0.725 ± 0.002 + + = +
LAA_B	0.719 ± 0.002 + - - =	0.703 ± 0.004 + = - =	0.712 ± 0.002 + = - =
ASA ($a = 1, m = T$)	0.734 ± 0.005 + = + +	0.721 ± 0.010 + + + +	0.724 ± 0.004 + + = +
ASA ($a = 5, m = T$)	0.745 ± 0.005 + + + +	0.723 ± 0.010 + + + +	0.722 ± 0.006 + + = +
ASA ($a = 10, m = T$)	0.737 ± 0.003 + = + +	0.703 ± 0.011 + = - =	0.708 ± 0.007 + - - =
ASA ($a = 50, m = T$)	0.693 ± 0.003 + - - -	0.643 ± 0.004 + - - -	0.639 ± 0.006 + - - -
ASA ($a = 100, m = T$)	0.672 ± 0.004 + - - -	0.620 ± 0.005 + - - -	0.607 ± 0.006 + - - -
ASA ($a = 500, m = T$)	0.638 ± 0.004 + - - -	0.599 ± 0.008 + - - -	0.563 ± 0.005 - - - -
ASA ($a = 1000, m = T$)	0.635 ± 0.004 + - - -	0.594 ± 0.010 + - - -	0.554 ± 0.003 - - - -
ASA ($a = 1, m = F$)	0.717 ± 0.007 + - - =	0.691 ± 0.013 + - - -	0.699 ± 0.008 + - - -
ASA ($a = 5, m = F$)	0.755 ± 0.004 + + + +	0.730 ± 0.008 + + + +	0.735 ± 0.005 + + + +
ASA ($a = 10, m = F$)	**0.761** ± 0.003 + + + +	**0.735** ± 0.015 + + + +	**0.742** ± 0.006 + + + +
ASA ($a = 50, m = F$)	0.749 ± 0.004 + + + +	0.673 ± 0.005 + - - -	0.699 ± 0.009 + - - -
ASA ($a = 100, m = F$)	0.717 ± 0.003 + - - -	0.645 ± 0.006 + - - -	0.664 ± 0.005 + - - -
ASA ($a = 500, m = F$)	0.665 ± 0.002 + - - -	0.621 ± 0.007 + - - -	0.621 ± 0.004 + - - -
ASA ($a = 1000, m = F$)	0.653 ± 0.003 + - - -	0.619 ± 0.007 + - - -	0.613 ± 0.002 + - - -
AUC			
	6HumanCoded	Sanders	SemEval
EAA_U	0.805 ± 0.005 = = - -	0.800 ± 0.017 = = + +	0.802 ± 0.006 = + - -
EAA_B	0.809 ± 0.001 = = = =	0.795 ± 0.016 = = + +	0.798 ± 0.007 - = - -
LAA_U	0.809 ± 0.001 + = = =	0.778 ± 0.002 - - = =	0.814 ± 0.000 + + = =
LAA_B	0.809 ± 0.001 + = = =	0.778 ± 0.003 - - = =	0.813 ± 0.001 + + = =
ASA ($a = 1, m = T$)	0.806 ± 0.003 = = - -	0.786 ± 0.007 - - + +	0.808 ± 0.002 + + - -
ASA ($a = 5, m = T$)	0.809 ± 0.002 = = = =	0.787 ± 0.005 - = + +	0.810 ± 0.003 + + - -
ASA ($a = 10, m = T$)	0.804 ± 0.001 = - - -	0.776 ± 0.008 - - = =	0.806 ± 0.003 + + - -
ASA ($a = 50, m = T$)	0.756 ± 0.003 - - - -	0.697 ± 0.005 - - - -	0.763 ± 0.002 - - - -
ASA ($a = 100, m = T$)	0.729 ± 0.002 - - - -	0.672 ± 0.006 - - - -	0.739 ± 0.002 - - - -
ASA ($a = 500, m = T$)	0.696 ± 0.003 - - - -	0.642 ± 0.008 - - - -	0.707 ± 0.005 - - - -
ASA ($a = 1000, m = T$)	0.690 ± 0.004 - - - -	0.637 ± 0.008 - - - -	0.701 ± 0.006 - - - -
ASA ($a = 1, m = F$)	0.793 ± 0.005 - - - -	0.762 ± 0.016 - - - -	0.787 ± 0.007 - - - -
ASA ($a = 5, m = F$)	0.837 ± 0.004 + + + +	0.807 ± 0.010 = = + +	0.833 ± 0.003 + + + +
ASA ($a = 10, m = F$)	**0.845** ± 0.001 + + + +	**0.812** ± 0.015 + + + +	**0.840** ± 0.003 + + + +
ASA ($a = 50, m = F$)	0.815 ± 0.003 + + + +	0.759 ± 0.006 - - - -	0.810 ± 0.004 + + - -
ASA ($a = 100, m = F$)	0.781 ± 0.003 - - - -	0.720 ± 0.007 - - - -	0.779 ± 0.004 - - - -
ASA ($a = 500, m = F$)	0.723 ± 0.002 - - - -	0.670 ± 0.008 - - - -	0.729 ± 0.005 - - - -
ASA ($a = 1000, m = F$)	0.712 ± 0.002 - - - -	0.665 ± 0.007 - - - -	0.721 ± 0.005 - - - -

Table 4. Macro-averaged F1 and AUC measures for different distant supervision models. Best results per column for each measure are given in bold.

and but clauses. We attribute these problems to two factors: 1) the annotation phase is solely based on unigrams, and 2) the current feature space omits the order in which words occur. The first factor could be addressed by using a lexicon of sentiment annotated phrases, and the second one by using more sophisticated feature representations such as n-grams or paragraph vector-embeddings [11].

We also study the effect of increasing the source corpus size in all different distant supervision methods: EAA, EAA_B, LAA, LAA_B, and ASA. It is important to remark that the number of generated instances in the four distant supervision baselines increases when increasing the size of the source corpus. The increments are proportional to the percentages shown in Table 3.

We trained classifiers using partitions of the source corpus ranging from ten thousand to ten million tweets. For the ASA model we set a to 10 and m to false, which were the best parameters according to the previous experiments (Table 4), and kept p and n with values set to $0.005 \times |\mathcal{C}|$, for generating balanced datasets with size equal to 1% of the size of the source corpus. Thus, the number of generated instances in ASA is also increased when using a larger source corpus.

The learning curves produced by logistic regressions applied to the SemEval dataset, trained with data generated using ASA and the four baselines from source corpora of different sizes, are shown in Figure 2. The performance metrics are again the macro-averaged F1 measure and AUC.

The figure indicates that most methods increase their performance when increasing the corpus size, and that these improvements tend to plateau when using more than 2 million tweets as input. We observe again that EAA exhibits poor performance in F1 and that balancing this method (EAA_B) produces substantial improvemets for this measure. Surprinsingly, the lexicon-based baselines LAA and LAA_B exhibit a slight decrease in F1 when increasing the source corpus size after the million tweet mark.

We observe in the initial part of the curves that LAA and LAA_B are the best distant supervision methods for source corpora smaller than 1 million tweets. This suggests that the prior knowledge from the lexicon can be very useful with small collections of data. It is important to consider that the setup of ASA for this experiment generates very few examples when the source corpus is small. This can be easily changed by generating more training data when the source

	Negative Tweets	Positive Tweets
f(x)=neg	Can we just haw class **cancelled** tomorrow? Cause I really don't **want** to go to BCA 101. I'd rather eat worms....	Never start working on your **dreams** and goals tomorrow... tomorrow never comes.... if it means anything to U, ACT NOW! #getafterit
	I never had a **good** time, I sat by my bedside. With papers and poetry about Estella	Just did Spartacus 2.0 and sauna imma be **sore** tomorrow but so **worth** it
	I got tickets to the NC State game saturday and nobody to go with..	@patrishuhx7 I have English tomorrow but it honestly doesn't **bother** me for some reason. Rella always makes my day. Don't ask
f(x)=pos	**Wish** me **lucky** on the Cahsee tomorrow I'm **pretty** **nervous**	**Happy** Valentine's Day!!! @MAziing: Everyday is the 14th!
	I haven't talked to you since July 19 th and all you can say is So do you **like** Beyonce's new cd GTFO	Ground hog day is such a **good** film, Sunday is for food and films #sunday
	Being in Amsterdam this early on a friday morning is not my ideal, I just **want** to get home!	Going to see Kendrick Lamar with @Pea_Starks in jan :D

Table 5. Examples of tweets classified with ASA. Positive and negative words from AFINN are marked with blue and red colours respectively. The leftmost column indicates the classifier's output.

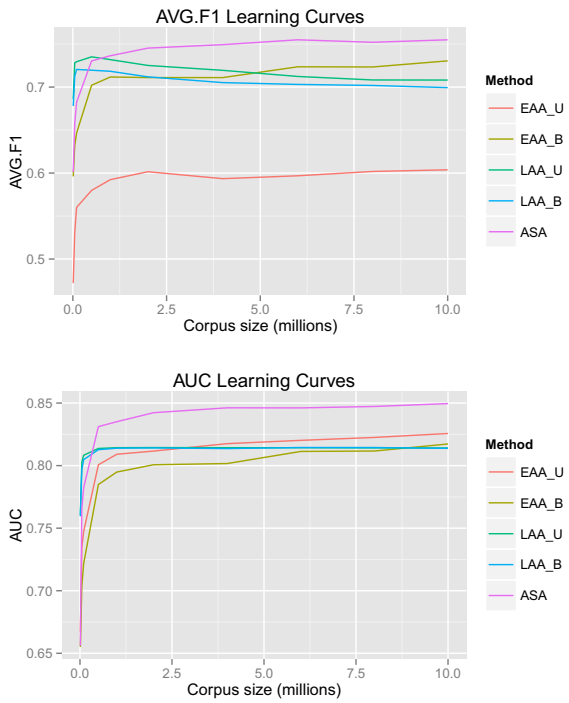

Figure 2. Learning curves over the SemEval dataset.

corpus is too small.

We also observe that after passing the million tweet mark, the emoticon-based models are better than LAA and LAA_B, and that ASA outperforms all the other models. These results indicate that ASA is a powerful distant supervision model that can be used for training accurate message-level polarity classifiers without relying on very large collections of unlabelled data.

6 Conclusions

We propose a new model called ASA to generate synthetic training data for Twitter sentiment analysis from unlabelled corpora using the prior knowledge provided by an opinion lexicon[15]. The method annotates tweets according to the polarity of their words, using a given polarity lexicon, and generates balanced training data by sampling and averaging tweets containing words with the same polarity. ASA is based on the lexical polarity hypothesis: because tweets are short messages, opinion words are strong indicators of the sentiment of the tweets in which they occur, and therefore tweets with at least one word with a certain known prior polarity are more likely to express the same polarity on the message level than the opposite one. The sample and average steps of ASA exploit this hypothesis by increasing the confidence of generating an instance located in the desired polarity region. ASA also incorporates a novel way for incorporating the knowledge provided by tweets with mixed positive and negative words.

The experimental results show that ASA produces better classifiers than the widely-adopted approach of using emoticons for labelling tweets into polarity classes and also better results than labelling tweets based on the polarity of their words, without sampling and averaging. Moreover, classifiers trained with data generated by ASA achieve better results than the other distant supervision models using substantially less training data. This shows that ASA can generate compact and efficient dataset for learning polarity concepts.

The proposed model can be used for training Twitter polarity classifiers in scenarios without labelled training data and for creating domain-specific sentiment classifiers by collecting data from the target domain. Considering that opinion lexicons are usually easier to obtain than corpora of polarity-annotated tweets, ASA can save significant labelling efforts for learning polarity classifiers in Twitter.

ASA opens several directions for further research. In essence, ASA allows the transfer of sentiment labels from the word-level to the message-level. Therefore, it could potentially be used for classifying tweets according to other sentiment labels associated with words, such as subjectivity labels, numerical scores indicating sentiment strength, and multi-label emotions.

Considering that ASA can generate large amounts of training data from large source corpora, it could also be suitable for training deep neural networks that learn more sophisticated representations of tweets for sentiment classification.

Another important aspect of ASA is its flexibility: it can be used with any kind of features for representing the tweets. For example, paragraph vector-embeddings [11], which have shown to be powerful representations for sentences, could be trained from large corpora of unlabelled tweets and included in the feature space.

Finally, ASA could also be adapted for training incremental polarity classifiers in an on-line fashion from a stream of time-evolving tweets. This approach could be used for online opinion mining from social media streams [2], and potentially be useful for tracking public opinion regarding high-impact events on Twitter, such as political campaigns, movie releases and natural disasters.

[15] The source code of the model is available for download at `http://www.cs.waikato.ac.nz/ml/sa/ds.html#asa`.

REFERENCES

[1] Finn Årup Nielsen, 'A new ANEW: Evaluation of a word list for sentiment analysis in microblogs', in *Proceedings of the ESWC2011 Workshop on 'Making Sense of Microposts': Big things come in small packages*, #MSM2011, pp. 93–98, (2011).

[2] Albert Bifet and Eibe Frank, 'Sentiment knowledge discovery in twitter streaming data', in *Proceedings of the 13th International Conference on Discovery science*, DS'10, pp. 1–15, Berlin, Heidelberg, (2010). Springer-Verlag.

[3] Felipe Bravo-Marquez, Marcelo Mendoza, and Barbara Poblete, 'Meta-level sentiment models for big social data analysis', *Knowledge-Based Systems*, **69**(0), 86 – 99, (2014).

[4] Peter F Brown, Peter V Desouza, Robert L Mercer, Vincent J Della Pietra, and Jenifer C Lai, 'Class-based n-gram models of natural language', *Computational Linguistics*, **18**(4), 467–479, (1992).

[5] Nitesh V. Chawla, Kevin W. Bowyer, Lawrence O. Hall, and W. Philip Kegelmeyer, 'Smote: Synthetic minority over-sampling technique', *J. Artif. Int. Res.*, **16**(1), 321–357, (June 2002).

[6] Alec Go, Richa Bhayani, and Lei Huang, 'Twitter sentiment classification using distant supervision', *CS224N Project Report, Stanford*, (2009).

[7] Nathalie Japkowicz and Shaju Stephen, 'The class imbalance problem: A systematic study', *Intell. Data Anal.*, **6**(5), 429–449, (October 2002).

[8] Soo-Min Kim and Eduard Hovy, 'Determining the sentiment of opinions', in *Proceedings of the 20th International Conference on Computational Linguistics*, pp. 1367–1373, Stroudsburg, PA, USA, (2004). Association for Computational Linguistics.

[9] Efthymios Kouloumpis, Theresa Wilson, and Johanna Moore, 'Twitter sentiment analysis: The good the bad and the omg!', *ICWSM*, **11**, 538–541, (2011).

[10] Krishna K. Ladha, 'Condorcet's jury theorem in light of de Finetti's theorem', *Social Choice and Welfare*, **10**(1), 69–85, (1993).

[11] Quoc V Le and Tomas Mikolov, 'Distributed representations of sentences and documents', in *Proceedings of the 31th International Conference on Machine Learning*, pp. 1188–1196, (2014).

[12] Shoushan Li, Zhongqing Wang, Guodong Zhou, and Sophia Yat Mei Lee, 'Semi-supervised learning for imbalanced sentiment classification', in *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence*, IJCAI'11, pp. 1826–1831. AAAI Press, (2011).

[13] Tao Li, Yi Zhang, and Vikas Sindhwani, 'A non-negative matrix tri-factorization approach to sentiment classification with lexical prior knowledge', in *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, ACL '09, pp. 244–252, Stroudsburg, PA, USA, (2009). Association for Computational Linguistics.

[14] Kun-Lin Liu, Wu-Jun Li, and Minyi Guo, 'Emoticon smoothed language models for twitter sentiment analysis', in *Proceedings of the National Conference on Artificial Intelligence*, pp. 1678–1684, (2012).

[15] Prem Melville, Wojciech Gryc, and Richard D. Lawrence, 'Sentiment analysis of blogs by combining lexical knowledge with text classification', in *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1275–1284, New York, NY, USA, (2009). ACM.

[16] Mike Mintz, Steven Bills, Rion Snow, and Dan Jurafsky, 'Distant supervision for relation extraction without labeled data', in *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, ACL '09, pp. 1003–1011, Stroudsburg, PA, USA, (2009). Association for Computational Linguistics.

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

[18] Saif M Mohammad, Svetlana Kiritchenko, and Xiaodan Zhu, 'NRC-canada: Building the state-of-the-art in sentiment analysis of tweets', in *Proceedings of the Seventh International Workshop on Semantic Evaluation Exercises*, SemEval'13, pp. 321–327, (2013).

[19] Preslav Nakov, Sara Rosenthal, Zornitsa Kozareva, Veselin Stoyanov, Alan Ritter, and Theresa Wilson, 'Semeval-2013 task 2: Sentiment analysis in twitter', in *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pp. 312–320, Atlanta, Georgia, USA, (June 2013). Association for Computational Linguistics.

[20] Alexander Pak and Patrick Paroubek, 'Twitter as a corpus for sentiment analysis and opinion mining', in *Proceedings of the Seventh International Conference on Language Resources and Evaluation*, pp. 1320–1326, Valletta, Malta, (2010).

[21] Saša Petrović, Miles Osborne, and Victor Lavrenko, 'The edinburgh twitter corpus', in *Proceedings of the NAACL HLT 2010 Workshop on Computational Linguistics in a World of Social Media*, WSA '10, pp. 25–26, Stroudsburg, PA, USA, (2010). Association for Computational Linguistics.

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

[23] Aliaksei Severyn and Alessandro Moschitti, 'Twitter sentiment analysis with deep convolutional neural networks', in *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 959–962, New York, NY, USA, (2015). ACM.

[24] Vikas Sindhwani and Prem Melville, 'Document-word co-regularization for semi-supervised sentiment analysis', in *Proceedings of the 2008 Eighth IEEE International Conference on Data Mining*, pp. 1025–1030, Washington, DC, USA, (2008). IEEE Computer Society.

[25] Michael Speriosu, Nikita Sudan, Sid Upadhyay, and Jason Baldridge, 'Twitter polarity classification with label propagation over lexical links and the follower graph', in *Proceedings of the First Workshop on Unsupervised Learning in NLP*, pp. 53–63, Stroudsburg, PA, USA, (2011). Association for Computational Linguistics.

[26] Duyu Tang, Furu Wei, Bing Qin, Ting Liu, and Ming Zhou, 'Coooolll: A deep learning system for twitter sentiment classification', in *Proceedings of the 8th International Workshop on Semantic Evaluation*, pp. 208–212, Dublin, Ireland, (August 2014). Association for Computational Linguistics and Dublin City University.

[27] Mike Thelwall, Kevan Buckley, and Georgios Paltoglou, 'Sentiment strength detection for the social web.', *JASIST*, **63**(1), 163–173, (2012).

[28] Peter 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*, pp. 417–424, Stroudsburg, PA, USA, (2002). Association for Computational Linguistics.

[29] Lei Zhang, Riddhiman Ghosh, Mohamed Dekhil, Meichun Hsu, and Bing Liu, 'Combining lexicon-based and learning-based methods for twitter sentiment analysis', Technical report, Hewlett-Packard Development Company, L.P., (2011).