Don't Count, Predict! An Automatic Approach to Learning Sentiment Lexicons for Short Text

Duy Tin Vo and **Yue Zhang**
Singapore University of Technology and Design
08 Somapah Road, Singapore 487372
duytin_vo@mymail.sutd.edu.sg and yue_zhang@sutd.edu.sg

Abstract

We describe an efficient neural network method to automatically learn sentiment lexicons without relying on any manual resources. The method takes inspiration from the NRC method, which gives the best results in SemEval13 by leveraging emoticons in large tweets, using the PMI between words and tweet sentiments to define the sentiment attributes of words. We show that better lexicons can be learned by using them to predict the tweet sentiment labels. By using a very simple neural network, our method is fast and can take advantage of the same data volume as the NRC method. Experiments show that our lexicons give significantly better accuracies on multiple languages compared to the current best methods.

1 Introduction

Sentiment lexicons contain the sentiment polarity and/or the strength of words or phrases (Baccianella et al., 2010; Taboada et al., 2011; Tang et al., 2014a; Ren et al., 2016a). They have been used for both rule-based (Taboada et al., 2011) and unsupervised (Turney, 2002; Hu and Liu, 2004; Kiritchenko et al., 2014) or supervised (Mohammad et al., 2013; Tang et al., 2014b; Vo and Zhang, 2015) machine-learning-based sentiment analysis. As a result, constructing sentiment lexicons is one important research task in sentiment analysis.

Many approaches have been proposed to construct sentiment lexicons. Traditional methods manually label the sentiment attributes of words (Hu and Liu, 2004; Wilson et al., 2005; Taboada et al., 2011). One benefit of such lexicons is high quality. On the other hand, the methods are time-consuming, requiring language and domain expertise. Recently, statistical methods have been exploited to learn sentiment lexicons automatically (Esuli and Sebastiani, 2006; Baccianella et al., 2010; Mohammad et al., 2013). Such methods leverage knowledge resources (Bravo-Marquez et al., 2015) or labeled sentiment data (Tang et al., 2014a), giving significantly better coverage compared to manual lexicons.

Among the automatic methods, Mohammad et al. (2013) proposed to use tweets with emoticons or hashtags as training data. The main advantage is that such training data are abundant, and manual annotation can be avoided. Despite that emoticons or hashtags can be noisy in indicating the sentiment of a tweet, existing research (Go et al., 2009; Pak and Paroubek, 2010; Agarwal et al., 2011; Kalchbrenner et al., 2014; Ren et al., 2016b) has shown that effectiveness of such data when used to supervise sentiment classifiers.

Mohammad et al. (2013) collect sentiment lexicons by calculating pointwise mutual information (PMI) between words and emoticons. The resulting lexicons give the best results in a SemEval13 benchmark (Nakov et al., 2013). In this paper, we show that a better lexicon can be learned by directly optimizing the prediction accuracy, taking the lexicon as input and emoticon as the output. The correlation between our method and the method of Mohammad et al. (2013) is analogous to the "predicting" vs "counting" correlation between distributional and distributed word representations (Baroni et al., 2014).

We follow Esuli and Sebastiani (2006) in using two simple attributes to represent each sentiment word, and take inspiration from Mikolov et al. (2013) in using a very simple neural network for sentiment prediction. The method can leverage the same data as Mohammad et al. (2013) and therefore benefits from both scale and annotation independence. Experiments show

Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, pages 219–224,
Berlin, Germany, August 7-12, 2016. ©2016 Association for Computational Linguistics

that the neural model gives the best results on standard benchmarks across multiple languages. Our code and lexicons are publicly available at *https://github.com/duytinvo/acl2016*.

2 Related work

Existing methods for automatically learning sentiment lexicons can be classified into three main categories. The first category augments existing lexicons with sentiment information. For example, Esuli and Sebastiani (2006) and Baccianella et al. (2010) use a tuple (pos, neg, neu) to represent each word, where pos, neg and neu stand for possibility, negativity and neutrality, respectively, training these attributes by extracting features from WordNet. These methods rely on the taxonomic structure of existing lexicons, which are limited to specific languages.

The second approach expands existing lexicons, which are typically manually labeled. For example, Tang et al. (2014a) apply a neural network to learn sentiment-oriented embeddings from a small amount of annotated tweets, and then expand a set of seed sentiment words by measuring vector space distances between words. Bravo-Marquez et al. (2015) extend an existing lexicon by classifying words using manual features. These methods are also limited to domains and languages with manual resources.

The third line of methods constructs lexicons from scratch by accumulating statistical information over large data. Turney (2002) proposes to estimate the sentiment polarity of words by calculating PMI between seed words and search hits. Mohammad et al. (2013) improve the method by computing sentiment scores using distance-supervised data from emoticon-baring tweets instead of seed words. This approach can be used to automatically extract multilingual sentiment lexicons (Salameh et al., 2015; Mohammad et al., 2015) without using manual resources, which makes it more flexible compared to the first two methods. We consider it as our baseline.

We use the same data source as Mohammad et al. (2013) to train lexicons. However, rather than relying on PMI, we take a machine-learning method in optimizing the prediction accuracy of emoticons using the lexicons. To leverage large data, we use a very simple neural network to train the lexicons.

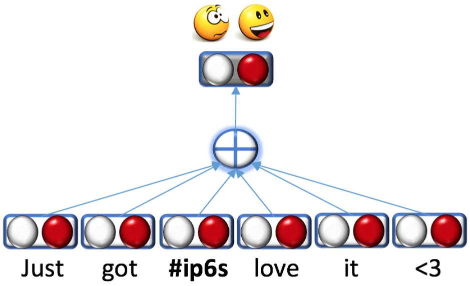

Figure 1: Our model.

3 Baseline

Mohammad et al. (2013) employ emoticons and relevant hashtags contained in a tweet as the sentiment label of the tweet. Given a set of tweets with their labels, the sentiment score (SS) for a token w was computed as:

$$SS(w) = PMI(w, pos) - PMI(w, neg), \quad (1)$$

where pos represents the positive label and neg represents the negative label. PMI stands for pointwise mutual information, which is

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

Here $freq(w, pos)$ is the number of times the term w occurs in positive tweets, $freq(w)$ is the total frequency of term w in the corpus, $freq(pos)$ is the total number of tokens in positive tweets, and N is the total number of tokens in the corpus. $PMI(w, neg)$ is calculated in a similar way. Thus, Equation 1 is equal to:

$$SS(w) = \log_2 \frac{freq(w, pos) * freq(neg)}{freq(w, neg) * freq(pos)} \quad (3)$$

4 Model

We follow Esuli and Sebastiani (2006), using positivity and negativity attributes to define lexicons. In particular, each word takes the form $w = (n, p)$, where n denotes negativity and p denotes positivity ($n, p \in \mathbf{R}$). As shown in Figure 1, given a tweet $tw = w_1, w_2, ..., w_n$, a simple neural network is used to predict its two-dimensional sentiment label y, where [1,0] for negative and [0,1] for positive tweets. The predicted sentiment probability y of a tweet is computed as:

$$h = \sum_i (w_i) \quad (4)$$

Language	#pos	#neg	#Tweets
English	4.5M	4.5M	9M
Arabic	400k	400k	800k

Table 1: Emoticon-based training data.

$$y = softmax(hW) \tag{5}$$

where W is fixed to the diagonal matrix ($W \in \mathbf{R}^{\mathbf{2x2}}$).

We follow Go et al. (2009) in defining the sentiment labels of tweets via emoticons. Each token is first initialized by random negative and positive attribute scores in [-0.25,0.25], and then trained by supervised learning. The cross-entropy error is employed as the objective function:

$$loss(tw) = -\sum \hat{y}.\log(y) \tag{6}$$

Backpropagation is applied to learn (n, p) for each token. Optimization is done using stochastic gradient descent over shuffled mini-batches, with the AdaDelta update rule (Zeiler, 2012). All models are trained over 5 epochs with a batch size of 50. Due to its simplicity, the method is very fast, training a sentiment lexicon over 9 million tweets within 35 minutes per epoch on an Intel® core™ i7-3770 CPU @ 3.40 GHz.

5 Sentiment Classification

The resulting lexicon can be used in both unsupervised and supervised sentiment classifiers. The former is implemented by summing the sentiment scores of all tokens contained in a given document (Taboada et al., 2011; Kiritchenko et al., 2014). If the total sentiment score is larger than 0, the document is classified as positive. Here only one positivity attribute is required to represent a lexicon, and we use the contrast between the positivity and negativity attributes ($p - n$) as the score.

The supervised method makes use of sentiment lexicons as features for machine learning classification. Given a document D, we follow Zhu et al. (2014) and extract the following features:

- The number of sentiment tokens in D, where sentiment tokens are word tokens whose sentiment scores are not zero in a lexicon;
- The total sentiment score of a document: $\sum_{w_i \in D} SS(w_i)$;
- The maximal score: $max_{w_i \in D} SS(w_i)$;
- The total scores of positive and negative words in D;
- The sentiment score of the last token in D.

Type		#pos	#neg	#Tweets
Supervised	train	3009	1187	4196
	dev	483	283	766
	test	1313	490	1803
Unsupervised		4805	1960	6765

Table 2: Statistics of the Semeval13.

Again we use $SS(w_i) = p_{w_i} - n_{w_i}$ as the sentiment score of each word w_i, because the methods are based on a single sentiment score value for each word.

6 Experiments

6.1 Experimental Settings

Training data: To automatically obtain training data, we use the Twitter Developers API[1] to crawl emoticon tweets[2] of English and Arabic from February 2014 to September 2014. We follow Go et al. (2009), removing all emoticons used to collect training data from the tweets, and Tang et al. (2014b), ignoring tweets which are less than 7 tokens. A Twitter tokenizer (Gimpel et al., 2011) is applied to preprocess all tweets. Rare words that occur less than 5 times in the vocabulary are removed. HTTP links and username are replaced by $\langle http \rangle$ and $\langle user \rangle$, respectively. The statistics of training data is shown in Table 1.

Sentiment classifier: We use LibLinear[3] (Fan et al., 2008) as the supervised classifier on benchmark datasets. The parameter c is tuned by making a grid search (Hsu et al., 2003) on the accuracy of development set on the English dataset and five-fold cross validation on the Arabic dataset.

Evaluation: We follow Kiritchenko et al. (2014) in employing precision (P), recall (R) and F1 score (F) to evaluate unsupervised classification. We follow Hsu et al. (2003) and use accuracy (acc), the tuning criterion, to evaluate supervised classification.

Code and lexicons: We make the Python implementation of our models and the resulting sentiment lexicons available at *https://github.com/duytinvo/acl2016*

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

[2] :), :), :-), :D, =) for positive and :(, : (, :-(for negative

[3] https://www.csie.ntu.edu.tw/∼cjlin/liblinear/

Lexicons		Unsup			Sup
		P	R	F	Acc
WEKA	ED	61	55.9	55.4	73.8
	STS	66.4	52.5	47.7	73.7
HIT		**75.3**	73.3	74.1	78.5
NRC	Hashtag	70.3	71.4	70.8	77.4
	Emoticon	73.2	74.6	73.8	79.9
nnLexicon		74.4	**77.3**	**75.3**	**81.3**

Table 3: Results on SemEval13 (English).

Labels	Balanced			Unbalanced		
	train	dev	test	train	dev	test
#pos	481	159	159	481	159	159
#neg				1012	336	336
#mix				500	166	166
#obj				4015	1338	1338
#Tweets	1924	636	636	6008	1999	1999

Table 4: Standard splits of ASTD.

6.2 English Lexicons

The Twitter benchmark of SemEval13 (Nakov et al., 2013) is used as the English test set. In order to evaluate both unsupervised and supervised methods, we follow Tang et al. (2014b) and Kiritchenko et al. (2014), removing neutral tweets. The statistics is shown in Table 2. We compare our lexicon with the lexicons of NRC[4] (Mohammad et al., 2013), HIT[5] (Tang et al., 2014a) and WEKA[6] (Bravo-Marquez et al., 2015). As shown in Table 3, using the unsupervised sentiment classification method (unsup) in Section 5, our lexicon gives significantly better result in comparison with count-based lexicons of NRC. Under both settings, our lexicon yields the best results compared to other methods.

6.3 Arabic Lexicons

We employ the standard Arabic Twitter dataset ASTD (Nabil et al., 2015), which consists of about 10,000 tweets with 4 labels: objective (obj), negative (neg), positive (pos) and mixed subjective (mix). The standard splits of ASTD are shown in Table 4. We follow Nabil et al. (2015) by merging training and validating data for learning model. We compare our lexicon with only the lexicons of NRC[7] (Salameh et al., 2015), because the methods of Tang et al. (2014a) and Bravo-Marquez et al. (2015) depend on manual resources, which are not available. As shown in Table 5, our lexicon consistently gives the best performance on both the balanced and unbalanced datasets, showing the advantage of “predicting” over “counting”.

Lexicons		Balanced	Unbalanced
NRC	Hashtag	31.9	63.4
	Emoticon	31.4	65.3
nnLexicon		**33.3**	**66.5**

Table 5: Results on ASTD (Arabic).

Words	nnLexicon	NRC
bad	-1.122	-1.295
worse	-1.626	-1.417
worst	-2.256	-1.875
busy	-0.520	-0.003
busier	-0.609	0.106*
busiest	-1.254	-0.712
suitable	0.502	-0.040*
satisfy	0.570	-0.173*
lazy	-0.462	0.224*
scummy	-0.852	0.049*
old wine	0.453	0.552
old meat	-0.172	0.014*
strong memory	0.081	-0.083*
strong snowstorm	-0.554	0.182*

Table 6: Example sentiment scores, where * denotes incorrect polarity.

6.4 Analysis

Table 6 shows examples of our predicting-based lexicon and the counting-based lexicon of Mohammad et al. (2013). First, both lexicons can correctly reflect the strength of emotional words (e.g. *bad, worse, worst*), which demonstrates that our method can learn statistical relevance as effectively as PMI. Second, we find many cases where our lexicon gives the correct polarity (e.g. *suitable, lazy*) but the lexicon of Mohammad et al. (2013) does not. To quantitatively compare the lexicons, we calculated the accuracies of their polarities (i.e. sign) by using the manually-annotated lexicon of Hu and Liu (2004) as the gold standard. We take the intersection between the automatic lexicons and the lexicon of Hu and Liu (2004) as the test set, which contains 3270 words. The polarity accuracy of our lexicon is 78.2%, in contrast to 76.9% by the lexicon of Mohammad et al. (2013), demonstrating the relative strength of our method. Third, by having two attributes (n, p) instead of one, our lexicon is better in compositionality (e.g. $SS(strong\ memory) > 0$, $SS(strong\ snowstorm) < 0$).

[4] http://saifmohammad.com/WebPages/Abstracts/NRC-SentimentAnalysis.htm

[5] http://ir.hit.edu.cn/∼dytang/

[6] http://www.cs.waikato.ac.nz/∼fjb11/

[7] http://saifmohammad.com/WebPages/ArabicSA.html

7 Conclusion

We constructed a sentiment lexicon for short text automatically using an efficient neural network, showing that prediction-based training is better than counting-based training for learning from large tweets with emoticons. In standard evaluations, the method gave better accuracies across multiple languages compared to the state-of-the-art counting-based method.

References

Apoorv Agarwal, Boyi Xie, Ilia Vovsha, Owen Rambow, and Rebecca Passonneau. 2011. Sentiment analysis of twitter data. In *Proceedings of the Workshop on Languages in Social Media*, LSM '11, pages 30–38.

Stefano Baccianella, Andrea Esuli, and Fabrizio Sebastiani. 2010. Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In *Proceedings of LREC*, volume 10, pages 2200–2204.

Marco Baroni, Georgiana Dinu, and Germán Kruszewski. 2014. Don't count, predict! a systematic comparison of context-counting vs. context-predicting semantic vectors. In *Proceedings of ACL*, pages 238–247.

Felipe Bravo-Marquez, Eibe Frank, and Bernhard Pfahringer. 2015. Positive, negative, or neutral: learning an expanded opinion lexicon from emoticon-annotated tweets. In *Proceedings of IJCAI*, pages 1229–1235.

Andrea Esuli and Fabrizio Sebastiani. 2006. Sentiwordnet: A publicly available lexical resource for opinion mining. In *Proceedings of LREC*, volume 6, pages 417–422.

Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. 2008. Liblinear: A library for large linear classification. *The Journal of Machine Learning Research*, 9:1871–1874.

Kevin Gimpel, Nathan Schneider, Brendan O'Connor, Dipanjan Das, Daniel Mills, Jacob Eisenstein, Michael Heilman, Dani Yogatama, Jeffrey Flanigan, and A. Noah Smith. 2011. Part-of-speech tagging for twitter: Annotation, features, and experiments. In *Proceedings of ACL-HLT*, pages 42–47.

Alec Go, Richa Bhayani, and Lei Huang. 2009. Twitter sentiment classification using distant supervision. *CS224N Project Report, Stanford*, 1:12.

Chih-Wei Hsu, Chih-Chung Chang, Chih-Jen Lin, et al. 2003. A practical guide to support vector classification.

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of KDD*, KDD '04, pages 168–177.

Nal Kalchbrenner, Edward Grefenstette, and Phil Blunsom. 2014. A convolutional neural network for modelling sentences. In *Proceedings of ACL*, pages 655–665.

Svetlana Kiritchenko, Xiaodan Zhu, and Saif M. Mohammad. 2014. Sentiment analysis of short informal texts. *J. Artif. Intell. Res.*, 50:723–762.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Saif M. Mohammad, Svetlana Kiritchenko, and Xiaodan Zhu. 2013. Nrc-canada: Building the state-of-the-art in sentiment analysis of tweets. In *Proceedings of SemEval-2013*, June.

Saif M Mohammad, Mohammad Salameh, and Svetlana Kiritchenko. 2015. How translation alters sentiment. *Journal of Artificial Intelligence Research*, 54:1–20.

Mahmoud Nabil, Mohamed Aly, and Amir F Atiya. 2015. Astd: Arabic sentiment tweets dataset. In *Proceedings of EMNLP*, pages 2515–2519.

Preslav Nakov, Sara Rosenthal, Zornitsa Kozareva, Veselin Stoyanov, Alan Ritter, and Theresa Wilson. 2013. Semeval-2013 task 2: Sentiment analysis in twitter. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of SemEval-2013*, pages 312–320.

Alexander Pak and Patrick Paroubek. 2010. Twitter based system: Using twitter for disambiguating sentiment ambiguous adjectives. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, SemEval '10, pages 436–439.

Yafeng Ren, Yue Zhang, Meishan Zhang, and Donghong Ji. 2016a. Context-sensitive twitter sentiment classification using neural network. In *Proceedings of AAAI*.

Yafeng Ren, Yue Zhang, Meishan Zhang, and Donghong Ji. 2016b. Improving twitter sentiment classification using topic-enriched multi-prototype word embeddings. In *Proceedings of AAAI*.

Mohammad Salameh, Saif Mohammad, and Svetlana Kiritchenko. 2015. Sentiment after translation: A case-study on arabic social media posts. In *Proceedings of NAACL-HLT*, pages 767–777, May–June.

Maite Taboada, Julian Brooke, Milan Tofiloski, Kimberly Voll, and Manfred Stede. 2011. Lexicon-based methods for sentiment analysis. *Computational linguistics*, 37(2):267–307.

Duyu Tang, Furu Wei, Bing Qin, Ming Zhou, and Ting Liu. 2014a. Building large-scale twitter-specific sentiment lexicon : A representation learning approach. In *Proceedings of COLING*, pages 172–182, August.

Duyu Tang, Furu Wei, Nan Yang, Ming Zhou, Ting Liu, and Bing Qin. 2014b. Learning sentiment-specific word embedding for twitter sentiment classification. In *Proceedings of ACL*, pages 1555–1565, June.

Peter Turney. 2002. Thumbs up or thumbs down? semantic orientation applied to unsupervised classification of reviews. In *Proceedings of ACL*.

Duy-Tin Vo and Yue Zhang. 2015. Target-dependent twitter sentiment classification with rich automatic features. In *Proceedings of IJCAI*, pages 1347–1353, July.

Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. 2005. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of HLT-EMNLP*.

Matthew D Zeiler. 2012. Adadelta: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701*.

Xiaodan Zhu, Svetlana Kiritchenko, and Saif Mohammad. 2014. Nrc-canada-2014: Recent improvements in the sentiment analysis of tweets. In *Proceedings of SemEval-2014*, pages 443–447.