Improving Twitter Sentiment Analysis with Topic-Based Mixture Modeling and Semi-Supervised Training

Bing Xiang *
IBM Watson
1101 Kitchawan Rd
Yorktown Heights, NY 10598, USA
bingxia@us.ibm.com

Liang Zhou
Thomson Reuters
3 Times Square
New York, NY 10036, USA
l.zhou@thomsonreuters.com

Abstract

In this paper, we present multiple approaches to improve sentiment analysis on Twitter data. We first establish a state-of-the-art baseline with a rich feature set. Then we build a topic-based sentiment mixture model with topic-specific data in a semi-supervised training framework. The topic information is generated through topic modeling based on an efficient implementation of Latent Dirichlet Allocation (LDA). The proposed sentiment model outperforms the top system in the task of *Sentiment Analysis in Twitter* in SemEval-2013 in terms of averaged F scores.

1 Introduction

Social media, such as Twitter and Facebook, has attracted significant attention in recent years. The vast amount of data available online provides a unique opportunity to the people working on natural language processing (NLP) and related fields. Sentiment analysis is one of the areas that has large potential in real-world applications. For example, monitoring the trend of sentiment for a specific company or product mentioned in social media can be useful in stock prediction and product marketing.

In this paper, we focus on sentiment analysis of Twitter data (tweets). It is one of the challenging tasks in NLP given the length limit on each tweet (up to 140 characters) and also the informal conversation. Many approaches have been proposed previously to improve sentiment analysis on Twitter data. For example, Nakov et al. (2013) provide an overview on the systems submitted to one of the SemEval-2013 tasks, *Sentiment Analysis in Twitter*. A variety of features have been utilized for sentiment classification on tweets. They include lexical features (e.g. word lexicon), syntactic features (e.g. Part-of-Speech), Twitter-specific features (e.g. emoticons), etc. However, all of these features only capture local information in the data and do not take into account of the global higher-level information, such as topic information.

Two example tweets are given below, with the word "*offensive*" appearing in both of them.

- *Im gonna post something that might be* **offensive** *to people in Singapore.*
- *#FSU* **offensive** *coordinator Randy Sanders coached for Tennessee in 1st #BCS title game.*

Generally "*offensive*" is used as a negative word (as in the first tweet), but it bears no sentiment in the second tweet when people are talking about a football game. Even though some local contextual features could be helpful to distinguish the two cases above, they still may not be enough to get the sentiment on the whole message correct. Also, the local features often suffer from the sparsity problem. This motivates us to explore topic information explicitly in the task of sentiment analysis on Twitter data.

There exists some work on applying topic information in sentiment analysis, such as (Mei et al., 2007), (Branavan et al., 2008), (Jo and Oh, 2011) and (He et al., 2012). All these work are significantly different from what we propose in this work. Also they are conducted in a domain other than Twitter. Most recently, Si et al. (2013) propose a continuous Dirichlet Process Mixture model for Twitter sentiment, for the purpose of stock prediction. Unfortunately there is no evaluation on the accuracy of sentiment classification alone in that work. Furthermore, no standard training or test corpus is used, which makes comparison with other approaches difficult.

Our work is organized in the following way:

* This work was done when the author was with Thomson Reuters.

Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Short Papers), pages 434–439,
Baltimore, Maryland, USA, June 23-25 2014.

- We first propose a universal sentiment model that utilizes various features and resources. The universal model outperforms the top system submitted to the SemEval-2013 task (Mohammad et al., 2013), which was trained and tested on the same data. The universal model serves as a strong baseline and also provides an option for smoothing later.

- We introduce a topic-based mixture model for Twitter sentiment. The model is integrated in the framework of semi-supervised training that takes advantage of large amount of un-annotated Twitter data. Such a mixture model results in further improvement on the sentiment classification accuracy.

- We propose a smoothing technique through interpolation between universal model and topic-based mixture model.

- We also compare different approaches for topic modeling, such as cross-domain topic identification by utilizing data from newswire domain.

2 Universal Sentiment Classifier

In this section we present a universal topic-independent sentiment classifier to establish a state-of-the-art baseline. The sentiment labels are either positive, neutral or negative.

2.1 SVM Classifier

Support Vector Machine (SVM) is an effective classifier that can achieve good performance in high-dimensional feature space. An SVM model represents the examples as points in space, mapped so that the examples of the different categories are separated by a clear margin as wide as possible. In this work an SVM classifier is trained with LibSVM (Chang and Lin, 2011), a widely used toolkit. The linear kernel is found to achieve higher accuracy than other kernels in our initial experiments. The option of probability estimation in LibSVM is turned on so that it can produce the probability of sentiment class c given tweet x at the classification time, i.e. $P(c|x)$.

2.2 Features

The training and testing data are run through tweet-specific tokenization, similar to that used in the CMU Twitter NLP tool (Gimpel et al., 2011). It is shown in Section 5 that such customized tokenization is helpful. Here are the features that we use for classification:

- Word N-grams: if certain N-gram (unigram, bigram, trigram or 4-gram) appears in the tweet, the corresponding feature is set to 1, otherwise 0. These features are collected from training data, with a count cutoff to avoid overtraining.

- Manual lexicons: it has been shown in other work (Nakov et al., 2013) that lexicons with positive and negative words are important to sentiment classification. In this work, we adopt the lexicon from Bing Liu (Hu and Liu, 2004) which includes about 2000 positive words and 4700 negative words. We also experimented with the popular MPQA (Wilson et al., 2005) lexicon but found no extra improvement on accuracies. A short list of Twitter-specific positive/negative words are also added to enhance the lexicons. We generate two features based on the lexicons: total number of positive words or negative words found in each tweet.

- Emoticons: it is known that people use emoticons in social media data to express their emotions. A set of popular emoticons are collected from the Twitter data we have. Two features are created to represent the presence or absence of any positive/negative emoticons.

- Last sentiment word: a "sentiment word" is any word in the positive/negative lexicons mentioned above. If the last sentiment word found in the tweet is positive (or negative), this feature is set to 1 (or -1). If none of the words in the tweet is sentiment word, it is set to 0 by default.

- PMI unigram lexicons: in (Mohammad et al., 2013) two lexicons were automatically generated based on pointwise mutual information (PMI). One is *NRC Hashtag Sentiment Lexicon* with 54K unigrams, and the other is *Sentiment140 Lexicon* with 62K unigrams. Each word in the lexicon has an associated sentiment score. We compute 7 features based on each of the two lexicons: (1) sum of sentiment score; (2) total number of

positive words (with score $s > 1$); (3) total number of negative words ($s < -1$); (4) maximal positive score; (5) minimal negative score; (6) score of the last positive words; (7) score of the last negative words. Note that for the second and third features, we ignore those with sentiment scores between -1 and 1, since we found that inclusion of those weak subjective words results in unstable performance.

- PMI bigram lexicon: there are also 316K bigrams in the *NRC Hashtag Sentiment Lexicon*. For bigrams, we did not find the sentiment scores useful. Instead, we only compute two features based on counts only: total number of positive bigrams; total number of negative bigrams.

- Punctuations: if there exists exclamation mark or question mark in the tweet, the feature is set to 1, otherwise set to 0.

- Hashtag count: the number of hashtags in each tweet.

- Negation: we collect a list of negation words, including some informal words frequently observed in online conversations, such as "*dunno*" ("don't know"), "*nvr*" ("never"), etc. For any sentiment words within a window following a negation word and not after punctuations '.', ',', ';', '?', or '!', we reverse its sentiment from positive to negative, or vice versa, before computing the lexicon-based features mentioned earlier. The window size was set to 4 in this work.

- Elongated words: the number of words in the tweet that have letters repeated by at least 3 times in a row, e.g. the word "*goooood*".

3 Topic-Based Sentiment Mixture

3.1 Topic Modeling

Latent Dirichlet Allocation (LDA) (Blei et al., 2003) is one of the widely adopted generative models for topic modeling. The fundamental idea is that a document is a mixture of topics. For each document there is a multinomial distribution over topics, and a Dirichlet prior $Dir(\alpha)$ is introduced on such distribution. For each topic, there is another multinomial distribution over words. One of the popular algorithms for LDA model parameter estimation and inference is Gibbs sampling (Griffiths and Steyvers, 2004), a form of Markov Chain Monte Carlo. We adopt the efficient implementation of Gibbs sampling as proposed in (Yao et al., 2009) in this work.

Each tweet is regarded as one document. We conduct pre-processing by removing stop words and some of the frequent words found in Twitter data. Suppose that there are T topics in total in the training data, i.e. t_1, t_2, ..., t_T. The posterior probability of each topic given tweet x_i is computed as in Eq. 1:

$$P_t(t_j|x_i) = \frac{C_{ij} + \alpha_j}{\sum_{k=1}^{T} C_{ik} + T\alpha_j} \quad (1)$$

where C_{ij} is the number of times that topic t_j is assigned to some word in tweet x_i, usually averaged over multiple iterations of Gibbs sampling. α_j is the j-th dimension of the hyperparameter of Dirichlet distribution that can be optimized during model estimation.

3.2 Sentiment Mixture Model

Once we identify the topics for tweets in the training data, we can split the data into multiple subsets based on topic distributions. For each subset, a separate sentiment model can be trained. There are many ways of splitting the data. For example, K-means clustering can be conducted based on the similarity between the topic distribution vectors or their transformed versions. In this work, we assign tweet x_i to cluster j if $P_t(t_j|x_i) > \tau$ or $P_t(t_j|x_i) = \max_k P_t(t_k|x_i)$. Note that this is a soft clustering, with some tweets possibly assigned to multiple topic-specific clusters. Similar to the universal model, we train T topic-specific sentiment models with LibSVM.

During classification on test tweets, we run topic inference and sentiment classification with multiple sentiment models. They jointly determine the final probability of sentiment class c given tweet x_i as the following in a sentiment mixture model:

$$P(c|x_i) = \sum_{j=1}^{T} P_m(c|t_j, x_i) P_t(t_j|x_i) \quad (2)$$

where $P_m(c|t_j, x_i)$ is the probability of sentiment c from topic-specific sentiment model trained on topic t_j.

3.3 Smoothing

Additionally, we also experiment with a smoothing technique through linear interpolation between the universal sentiment model and topic-based sentiment mixture model.

$$P(c|x_i) = \theta \times P_U(c|x_i) + (1-\theta) \times \sum_{j=1}^{T} P_m(c|t_j, x_i) P_t(t_j|x_i) \quad (3)$$

where θ is the interpolation parameter and $P_U(c|x_i)$ is the probability of sentiment c given tweet x_i from the universal sentiment model.

4 Semi-supervised Training

In this section we propose an integrated framework of semi-supervised training that contains both topic modeling and sentiment classification. The idea of semi-supervised training is to take advantage of large amount low-cost un-annotated data (tweets in this case) to further improve the accuracy of sentiment classification. The algorithm is as follows:

1. Set training corpus D for sentiment classification to be the annotated training data D_a;
2. Train a sentiment model with current training corpus D;
3. Run sentiment classification on the unannotated data D_u with the current sentiment model and generate probabilities of sentiment classes for each tweet, $P(c|x_i)$;
4. Perform data selection. For those tweets with $P(c|x_i) > p$, add them to current training corpus D. The rest is used to replace the unannotated corpus D_u;
5. Train a topic model on D, and store the topic inference model and topic distributions of each tweet;
6. Cluster data in D based on the topic distributions from Step 5 and train a separate sentiment model for each cluster. Replace current sentiment model with the new sentiment mixture model;
7. Repeat from Step 3 until finishing a predetermined number of iterations or no more data is added to D in Step 4.

5 Experimental Results

5.1 Data and Evaluation

We conduct experiments on the data from the task B of *Sentiment Analysis in Twitter* in SemEval-2013. The distribution of positive, neutral and negative data is shown in Table 1. The development set is used to tune parameters and features. The test set is for the blind evaluation.

Set	Pos	Neu	Neg	Total
Training	3640	4586	1458	9684
Dev	575	739	340	1654
Test	1572	1640	601	3813

Table 1: Data from SemEval-2013. Pos: positive; Neu: neutral; Neg: negative.

For semi-supervised training experiments, we explored two sets of additional data. The first one contains 2M tweets randomly sampled from the collection in January and February 2014. The other contains 74K news documents with 50M words collected during the first half year of 2013 from online newswire.

For evaluation, we use macro averaged F score as in (Nakov et al., 2013), i.e. average of the F scores computed on positive and negative classes only. Note that this does not make the task a binary classification problem. Any errors related to neutral class (false positives or false negatives) will negatively impact the F scores.

5.2 Universal Model

In Table 2, we show the incremental improvement in adding various features described in Section 2, measured on the test set. In addition to the features, we also find SVM weighting on the training samples is helpful. Due to the skewness in class distribution in the training set, it is observed during error analysis on the development set that subjective (positive/negative) tweets are more likely to be classified as neutral tweets. The weights for positive, neutral and negative samples are set to be (1, 0.4, 1) based on the results on the development set. As shown in Table 2, weighting adds a 2% improvement. With all features combined, the universal sentiment model achieves 69.7 on average F score. The F score from the best system in SemEval-2013 (Mohammad et al., 2013) is also listed in the last row of Table 2 for a comparison.

Model	Avg. F score
Baseline with word N-grams	55.0
+ tweet tokenization	56.1
+ manual lexicon features	62.4
+ emoticons	62.8
+ last sentiment word	63.7
+ PMI unigram lexicons	64.5
+ hashtag counts	65.0
+ SVM weighting	67.0
+ PMI bigram lexicons	68.2
+ negations	69.0
+ elongated words	69.7
Mohammad et al., 2013	69.0

Table 2: Results on the test set with universal sentiment model.

5.3 Topic-Based Mixture Model

For the topic-based mixture model and semi-supervised training, based on the experiments on the development set, we set the parameter τ used in soft clustering to 0.4, the data selection parameter p to 0.96, and the interpolation parameter for smoothing θ to 0.3. We found no more noticeable benefits after two iterations of semi-supervised training. The number of topics is set to 100.

The results on the test set are shown Table 3, with the topic information inferred from either Twitter data (second column) or newswire data (third column). The first row shows the performance of the universal sentiment model as a baseline. The second row shows the results from re-training the universal model by simply adding tweets selected from two iterations of semi-supervised training (about 100K). It serves as another baseline with more training data, for a fair comparison with the topic-based mixture modeling that uses the same amount of training data.

We also conduct an experiment by only considering the most likely topic for each tweet when computing the sentiment probabilities. The results show that the topic-based mixture model outperforms both the baseline and the one that considers the top topics only. Smoothing with the universal model adds further improvement in addition to the un-smoothed mixture model. With the topic information inferred from Twitter data, the F score is 2 points higher than the baseline without semi-supervised training and 1.4 higher than the baseline with semi-supervised data.

Model	Tweet-topic	News-topic
Baseline	69.7	69.7
+ semi-supervised	70.3	70.2
top topic only	70.6	70.4
mixture	71.2	70.8
+ smoothing	71.7	71.1

Table 3: Results of topic-based sentiment mixture model on SemEval test set.

As shown in the third column in Table 3, surprisingly, the model with topic information inferred from the newswire data works well on the Twitter domain. A 1.4 points of improvement can be obtained compared to the baseline. This provides an opportunity for cross-domain topic identification when data from certain domain is more difficult to obtain than others.

In Table 4, we provide some examples from the topics identified in tweets as well as the newswire data. The most frequent words in each topic are listed in the table. We can clearly see that the topics are about phones, sports, sales and politics, respectively.

Tweet-1	Tweet-2	News-1	News-2
phone	game	sales	party
call	great	stores	government
answer	play	online	election
question	team	retail	minister
service	win	store	political
text	tonight	retailer	prime
texting	super	business	state

Table 4: The most frequent words in example topics from tweets and newswire data.

6 Conclusions

In this paper, we presented multiple approaches for advanced Twitter sentiment analysis. We established a state-of-the-art baseline that utilizes a variety of features, and built a topic-based sentiment mixture model with topic-specific Twitter data, all integrated in a semi-supervised training framework. The proposed model outperforms the top system in SemEval-2013. Further research is needed to continue to improve the accuracy in this difficult domain.

References

David Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent Dirichlet allocation. *In Journal of Machine Learning Research.* 3(2003), 993–1022.

S. R. K. Branavan, Harr Chen, Jacob Eisenstein, and Regina Barzilay. 2008. Learning document-level semantic properties from free-text annotations. *In Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL-2008).*

Chih-Chung Chang and Chih-Jen Lin. 2011. LIBSVM: A library for support vector machines. *In ACM Transactions on Intelligent Systems and Technology.*

Kevin Gimpel, Nathan Schneider, Brendan O'Connor, Dipanjan Das, Daniel Mills, Jacob Eisenstein, Michael Heilman, Dani Yogatama, Jeffrey Flanigan, and Noah A. Smith. 2011. Part-of-speech tagging for Twitter: annotation, features, and experiments. *In Proceedings of the Annual Meeting of the Association for Computational Linguistics.*

Thomas L. Griffiths and Mark Steyvers. 2004. Finding scientific topics. *In Proceedings of the National Academy of Science.* 101, 5228–5235.

Yulan He, Chenghua Lin, Wei Gao, and Kam-Fai Wong. 2012. Tracking sentiment and topic dynamics from social media. *In Proceedings of the 6th International AAAI Conference on Weblogs and Social Media (ICWSM-2012).*

Mingqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. *In Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.*

Yohan Jo and Alice Oh. 2011. Aspect and sentiment unification model for online review analysis. *In Proceedings of ACM Conference in Web Search and Data Mining (WSDM-2011).*

Qiaozhu Mei, Xu Ling, Matthew Wondra, Hang Su, and ChengXiang Zhai. 2007. Topic sentiment mixture: modeling facets and opinions in weblogs. *In Proceedings of International Conference on World Wide Web (WWW-2007).*

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 the Seventh International Workshop on Semantic Evaluation (SemEval 2013).* 312-320, Atlanta, Georgia, June 14-15, 2013.

Preslav Nakov, Zornitsa Kozareva, Alan Ritter, Sara Rosenthal, Veselin Stoyanov, and Theresa Wilson. 2013. SemEval-2013 Task 2: Sentiment Analysis in Twitter. *In Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013).* 312-320, Atlanta, Georgia, June 14-15, 2013.

Jianfeng Si, Arjun Mukherjee, Bing Liu, Qing Li, Huayi Li, and Xiaotie Deng. 2013. Exploiting topic based Twitter sentiment for stock prediction. *In Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics.* 24-29, Sofia, Bulgaria, August 4-9,2013.

Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. 2005. Recognizing contextual polarity in phrase-level sentiment analysis. *In Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing, HLT 05.*

Limin Yao, David Mimno, and Andrew McCallum. 2009. Efficient methods for topic model inference on streaming document collections. *KDD'09.*