Automatic Sentiment Monitoring of Specific Topics in the Blogosphere

Fernanda S. Pimenta, Darko Obradović, Rafael Schirru,
Stephan Baumann, and Andreas Dengel

German Research Center for Artificial Intelligence (DFKI),
Knowledge Management Department
& University of Kaiserslautern,
Computer Science Department
Kaiserslautern & Berlin, Germany
{fpimenta,obradovic,schirru,baumann,dengel}@dfki.uni-kl.de

Abstract. The classification of a text according to its sentiment is a task of raising relevance in many applications, including applications related to monitoring and tracking of the blogosphere. The blogosphere provides a rich source of information about products, personalities, technologies, etc. The identification of the sentiment expressed in articles is an important asset to a proper analysis of this user-generated data. In this paper we focus on the task of automatic determination of the polarity of blogs articles, i. e., the sentiment analysis of blogs. In order to identify whether a piece of text expresses a positive or negative opinion, an approach based on word spotting was used. Empirical results on different domains show that our approach performs well if compared to costly and domain-specific approaches. In addition to that, if we consider an aggregation of a set of documents and not the polarity of each individual document, we can achieve an accuracy distribution around 90% for specific topics of a certain domain.

Keywords: opinion mining, sentiment analysis, blogosphere

1 Introduction

In order to achieve a better analysis and organization of the large amount of online documents available nowadays, it is very useful to classify texts according to the sentiment that they express [1]. The sentiment analysis of texts can be applied to various tasks such as text summarization, management of online forums, and monitoring of the acceptance of a given product or brand through the tracking of discussions on weblogs [2]. The blogosphere provides a rich source of information about products, personalities, technologies, etc. The identification of the sentiment expressed in blogs is an important asset to a proper analysis of this user-generated data.

Not only big companies benefit from sentiment analysis, but also politicians, journalists, advertisers, and market researchers. The research in this field encompasses diverse domains such as movies (e. g., [1],[3]), cars, books, travel (e. g., [4]),

and many other products and services (e. g., [5]). The large amount of available information sources and different domains make an automatic approach for the sentiment analysis of the blogosphere indispensable. In this paper, we focus on the problem of classifying a text according to its polarity, which can be one out of positive, negative, or neutral, in a non-domain-specific and scalable way. Some known methods were implemented based on word spotting for the realization of this task and performed an evaluation of them using datasets from different domains.

The remainder of this article is structured as follows. In Section 2, we present related work in the field of sentiment analysis. We describe the methods implemented for the classification of text according to its polarity in Section 3. Next, in Section 4 we perform an evaluation of the methods. In Section 5 we consider how sentiment analysis can be used to monitor the blogosphere considering an aggregation of articles in topics. Then we present our findings and our ideas for future work in Section 6.

2 Related Work

Words and expressions that compose a text possess an evaluative character that varies not only in degree, but also in polarity [6]. A positive polarity means a positive evaluation and a negative polarity means a negative evaluation. In the sentiment analysis field, a large amount of work focuses on the classification of text according to its polarity. Identifying whether a text is either positive, negative, or neutral usually is done with word spotting techniques or machine learning. Word spotting techniques rely on sentiment bearing words and expressions that are either present in an affective lexicon or have their sentiment captured by an automatic approach.

Turney and Littman [6] proposed a method to automatically predict the polarity score of a word or phrase by its statistic association with a set of negative and positive paradigm words. This strategy is called Semantic Orientation from Association (SO-A) . The SO-A of a word/phrase is calculated by the difference between its power of association with the set of positive and its power of association with the negative set. They used two different measures to calculate the association: pointwise mutual information (PMI) and latent semantic analysis (LSA). With a different idea, Pang et al. [1] applied machine learning techniques to perform sentiment analysis in movie reviews. They employed Naïve Bayes, maximum entropy classification, and support vector machines, and although not as good as for topic categorization, the results were satisfactory. Gamon [5] also successfully used machine learning for the classification of consumer reviews, and besides predicting whether a review was positive or negative, it established a ranking (from 1 to 4) on it. The author manage to improve his SVM approach by also taking into account the effects of valence shifters over words and expressions.

Nigam and Hurst [7] presented a system to automatically detect polar expressions about a given topic through the integration of a shallow NLP polar

language extraction system and a machine learning based topic classifier. The results of their experiments show that if considered separately, the polarity classifier performs better than when applied together with the topic classifier. In the field of weblogs, Durant and Smith [8] applied a Naïve Bayes classifier together with a forward feature selection technique to identify the political sentiment of weblog posts. Their classifier performed well (even outperforming SVM), but their focus was a little bit different. They aimed to predict the left or right political alignment of posts. A very similar work to the one of Durant and Smith [8], but with the same task as ours (identifying positive and negative sentiment in blogs), is the one presented by Melville et al. [2]. They introduced a framework which uses background lexical information together with supervised learning as an approach to sentiment classification. Their results show that the approach is a good alternative to reducing the burden of labeling many examples in the target domain. However, like many other machine learning approaches, their experiments rely on well-balanced and structured datasets, many times from a unique domain or topic. Besides that, the previously mentioned studies take into account the polarity of individual documents, not of an aggregation of a set of documents, an approach that is considered in this paper.

3 Sentiment Classification

The sentiment analysis of a text can be performed based on the sentiment bearing terms (words or expressions) that comprise such text, e. g., using word spotting techniques. Through the counting of terms it is possible to classify the text according to its polarity. Counting positive and negative terms is a very simple technique proposed in [4] and [9] and may well be used to classify entire documents. Different from the approaches based on machine learning, term counting does not require training and it is suitable even when training data is not available. If the majority of the sentiment bearing terms of a text is positive, the text is considered positive. Otherwise, if the majority of these terms is negative, the text is classified as negative. If there is some kind of balance between positive and negative terms, the text is considered neutral. Term counting relies on words and expressions that are either present in an affective lexicon or have their polarity captured by an automatic approach. We implemented these two types of term counting approaches and called them *lexicon based approach* and semantic orientation from association approach.

3.1 Lexicon Based Approach

First of all, we perform sentence segmentation and part-of-speech tagging (POS tagging) over the text we want to classify using the JTextPro text processing toolkit [10]. Then, for each of the terms considered sentiment relevant in the text, we consult an affective lexicon that contains polarity information about these terms. We have chosen *SentiWordNet* [11] as our affective lexicon since it is a lexical resource freely available for educational and research purposes.

We use here *SentiWordNet* 1.0 (the latest version available at the time of our experiments). Through the combination of the results produced by eight ternary classifiers, *SentiWordNet* associates for each of the *synsets* of *WordNet* (version 2.0) three scores related to polarity properties (positive, negative, and objective) that each ranges from -1 to 1. For this approach, identifying the polarity score of a text consists then in calculating the average polarity score of the terms that comprise it. We considered here two variant methods depending on which terms should be used in the calculation. In the first, only adjectives and adverbs of the sentences are taken into account (we call it LB_AdjAdv). The second one is a modification of LB_AdjAdv (we call it LB_AdjAdvMod), in which the effect of contextual valence shifters on the polarities of the adjectives and adverbs are considered. The concept of contextual valence shifters was introduced in [3]. They consist of negations, intensifiers and diminishers and they flip, increase, or decrease the polarity score of a sentiment term. When either an adjective or an adverb is found, we look for contextual valence shifters that occur near it and, if found, the weights of the valence shifters are multiplied with the original score of the adjective/adverb. Table 1 shows an example of the impact of valence shifters on the word *cool*, which originally has the positive polarity score of 0.5 according to *SentiWordNet*.

Table 1. Example of the effect of valence shifters over the word *cool*.

Valence Shifter	**Score**
None	0.5
Negation (e. g., not)	-0.5
Intensifier(e. g., very)	1.0
Diminisher(e. g., slightly)	0.25

3.2 Semantic Orientation from Association Approach

Like in the above mentioned approach, we first segment the text and then apply POS-tagging on it using the JTextPro text processing toolkit [10]. Second, we use patterns of POS tags defined in [4] for extracting phrases from the processed text (Table 2). The JJ tags are adjectives, the NN tags are nouns, the RB tags are adverbs, and the VB tags are verbs [1]. For each phrase, we then calculate the SO-A of it using as the measure of association the Pointwise Mutual Information (SO-PMI). Based on [6], in order to calculate the PMI of each phrase, we issue queries to a search engine (in our case, Yahoo![2]) and count the number of hits the set of paradigm words gets alone and the number of hits it gets with the phrase. Let $Pwords$ be the set of paradigm positive words and $Nwords$ the set

[1] For a complete reference on the POS tags, see [12]
[2] Using the Yahoo! API available at http://developer.yahoo.com/

of paradigm negative words. The SO-PMI of a phrase, i. e., its polarity score, is defined as

$$SO - PMI(phrase) = \log_2 \frac{hits(phrase, Pwords)hits(Nwords)}{hits(phrase, Nwords)hits(Pwords)}$$

Table 2. Patterns of POS tags for extracting two-Word Phrases [4].

	First Word	Second Word	Third Word (not extracted)
1.	JJ	NN or NNS	anything
2.	RB, RBR, or RBS	JJ	not NN nor NNS
3.	JJ	JJ	not NN nor NNS
4.	NN or NNS	JJ	not NN nor NNS
5.	RB, RBR or RBS	VB, VBD, VBN, or VBG	anything

The polarity of the entire text is then calculated by the average of the SO-PMI scores of all the phrases that comprise it. We call this method SO_PMI.

4 Evaluation

In this section, we present experiments and analyses of the application of the implemented methods. We perform two sets of experiments. The first compares all the methods implemented and choose the best of them. The comparison of all methods is only performed with one data set because of time limitations to execute the SO_PMI method. The API used to issue queries to the Yahoo! search engine has a limit of 5000 queries a day, and to calculate the SO-PMI of all the phrases in all the data sets, would take a quite long time (around forty days).

4.1 Data sets

Our motivating application is to perform the sentiment analysis of blog posts. Blogs are much more diverse and complex in structure than reviews. However, since there is a great amount of sentiment annotated data sets regarding reviews and they have been used extensively in previous sentiment analysis works, we decided also to use these data sets in our empirical evaluation. We have used the following publicly available data sets.

Amazon Reviews The data set for the first set of experiments is comprised of 1000 Amazon camera and photo product reviews and it was first presented in [13]. Each review consists of a rating that ranges from 1 to 5 stars. Reviews with rating values greater than 3 were labeled as positive, those with rating values of

less than 3 were labeled negative, and the rest discarded because their polarity was considered ambiguous. We make this assumption about the ratings based on previous works that have already used this dataset (e. g., [13] and [14]), although it is well known that users rate items with different personal scales and this issue should be considered when estimating the relevance of items for a certain user [15]. In the end we have 500 positives and 500 negatives reviews for this dataset.

Convote This data set was introduced in [16] and consists of automatically transcribed political debates classified according to whether an utterance is in support of a motion, or in opposition to it. There were in total 701 utterance, 426 in support and 275 in opposition.

Movie Reviews Provided by [1], this data consists of 1000 positive and 1000 negative reviews from the Internet Movie Database. Positive labels were assigned to reviews that had a rating above 3.5 stars and negative labels were assigned to the rest. We use version 2.0 of this dataset in our experiments.

Service reviews This data set contains reviews of six different domains and was provided by Whitehead and Yaeger [17]. The domains, as well the amount of positive and negative reviews of each domain are summarized in Table 3

Table 3. Domains of the Whitehead and Yaeger [17] data set

Domain	Positive reviews	Negative reviews	Total
Camp	402	402	804
Doctor	739	739	1478
Drug	401	401	802
Lawyer	110	110	220
Radio	502	502	1004
Tv	235	235	470

4.2 Results

We carried out two sets of experiments, one with the Amazon reviews data set and the other with the remaining data sets. In the first set of experiments, we used the accuracy of the classification in order to determine which approach works best on the data set. We present in Table 4 the results of these experiments based on the accuracy of classifying the reviews correctly (as either positive or negative), i. e., the total number of reviews correctly classified against the total number of reviews.

Table 4. Comparing accuracy of different approaches to sentiment classification with the Amazon reviews data set.

Method	**Accuracy**
LB_AdjAdv	61%
LB_AdjAdvMod	63%
SO_PMI	51%

It can be seen in Table 1 that the accuracy for the LB_AdjAdvMod is the highest. Although there is no huge difference between LB_AdjAdv and LB_AdjAdvMod, the addition of contextual valence shifters improves the accuracy of classification, as already shown in [3]. The surprise here was the poor performance of the method using SO_PMI. Using SO-PMI, Turney and Littman [6] obtained in their experiments an accuracy around 80% to automatically predict the polarity score of words. In our experiments, the accuracy of this method is as good as a random classifier (that would achieve 50% of accuracy). This is probably due to the fact that the scores computed with SO-PMI are not always trustworthy. One possible problem is that the number of hits returned by a search engine is not known to be 100% reliable and hence the calculation of the SO-PMI of the phrases would not be 100% reliable too.

Since the LB_AdjAdvMod method was the best in the first set of experiments, we choose it to be used as the classifier for the second set of experiments. We performed the classification on the rest of the non-blog data sets and the results concerning accuracy are shown in Table 5.

Table 5. Accuracy of LB_AdjAdvMod approach in different data sets.

Data Set	**Accuracy**
Convote	55%
Movie Reviews	60%
Camp	66%
Doctor	72%
Drug	61%
Lawyer	74%
Radio	61%
Tv	63%

The results demonstrate that for all data sets, the classifier performs better than a random classifier (with a baseline of 50%). The algorithm achieves an accuracy of 74% with the Doctor data set which is satisfactory compared to the methods that exist so far. However, for the Convote and Movie Reviews data sets the results are still very close to the random classifier. The poor results for

the Convote data set may be related to the fact that it consists of transcribed spoken political debates and not originally written text. This could influence the performance of the classifier since it was created aiming at written language, not spoken. On the other hand, for the Movie Reviews data set, maybe the problem was the fact that sometimes a review contains negative words describing the plot of the movie, but this does not mean that the review is negative [3].

5 Sentiment Monitoring of Topics

The accuracy of sentiment analysis is still not satisfactory when compared with other automatic classifiers. Natural language is highly complex, the state of the art not reliable, and some critics doubt it will ever work since this task is difficult even for humans. However, it is possible to use sentiment analysis to monitor the distribution of polarity over a set of documents of a specific topic instead of individual documents. Our hypothesis is to consider the distribution of polarity over an aggregation of documents in order to achieve much more reliable results with today's mediocre classifier accuracies. In order to analyze this idea, we performed a new set of experiments with the LB_AdjAdvMod method using a data set comprised of blog articles from different topics of a given domain.

5.1 Android blogs data set

To test our best approach in the domain of blogs, we have annotated a set of blog articles with sentiment scores. The original blog data collection used here was presented in Schirru et al. [18] and comprises blog articles categorized into topics. Per topic we read each article and annotated it manually as either positive, negative, or neutral. Table 6 shows the topics that comprise the final labeled set.

Table 6. Topics of the Android blogs data set.

Topic	Number of Articles
cupcake	118
dev-phone-block	77
htc-magic	68
uk-app-market	54
amazon-deal	48
robot-control	48
windows-mobile	24
gartner-study	17
iverse-comics	14

5.2 Evaluation

We classified the articles in the Android blogs data set using the LB_AdjAdvMod method and compared the resulting classification with the manually created ground truth (GT). The distribution of polarity over the set of articles of each specific topic was used then as an initial evaluation. Considering an interval from -0.1 to 0.1 for the neutral class, we aggregated the articles according to their polarities in three classes: negative, neutral, and positive. The difference between the total number of articles in each GT class and the total number of articles in our classifier's class is calculated. Then, we calculate the penalty cost to equalize the LB_AdjAdvMod distribution with the GT distribution. Considering that the cost to transfer one article from neutral to any of the other classes (or vice-versa) is 0.5, and from positive to negative (or vice-versa) is 1, the accuracy distribution of our classifier will be the total penalty cost divided by the total number of articles of the topic. As a baseline, we take the classification of a random classifier (RC) that distributes evenly the articles among the three polarity classes. For this classifier, the worst case is when all the articles in the GT belong to the positive class (or the negative class). However, even in the worst case, the accuracy distribution of the RC will never be lower than 0.5. In Table 7, we have the distributions for the topic *dev-phone-block*. This topic

Table 7. Distribution of the *dev-phone-block* articles into polarity classes according to three classifiers

Classifier	**Negative**	**Neutral**	**Positive**
GT	56	20	1
LB_AdjAdvMod	42	26	9
RC	25.67	25.67	25.67

concerns the announcement of the android market blocking some new merchant applications which caused the frustration of many developers. As we can see by the distributions of the LB_AdjAdvMod and the GT, the classifier captures well the tendency of the overall sentiment towards the topic (mostly negative in this case). Calculating the accuracy distribution for the LB_AdjAdvMod method, we get 85.71% against 64.29% for the RC, showing that our method performs better than chance.

Table 8 shows the values for the accuracy distribution for the classification of the LB_AdjAdvMod method and the RC considering the GT of the Android blogs data set. For most of the topics our classification performs well, however, for a few of them it is as good as RC. Reading the articles from topics like *uk-app-market* and *amazon-deal* we can observe that there is no tendency to a more positive or more negative sentiment towards the topic. These articles are more objective and don't have good indications of sentiment.

Table 8. Accuracy distribution per topic of the LB_AdjAdvMod method and the RC for the Android blogs data set.

Topic	LB_AdjAdvMod	Random Classifier
cupcake	87.71%	66.81%
dev-phone-block	85.71%	64.29%
htc-magic	92.65%	83.33%
uk-app-market	70.37%	70.37%
amazon-deal	85.42%	86.46%
robot-control	80.21%	71.88%
windows-mobile	85.42%	81.25%
gartner-study	97.06%	74.51%
iverse-comics	89.29%	82.14%

6 Conclusion and Future Work

We have implemented methods for sentiment analysis using word spotting approaches. Empirical results on different domains show that although our best approach performs well if compared to costly and domain-specific approaches, it is still not satisfactory. However, if we consider the distribution of polarity over an agreggation of documents we have much more reliable results than considering the classification of each document separately. We analyzed this distribution in a set of articles of different topics of a certain domain and we noticed that our method can provide good indications for the sentiment monitoring of the blogosphere. We believe this method is also useful in domains where the number of positive and negative samples is not normally balanced (e. g., the movies domain).

Increasing the list of contextual valence shifters and using an affective lexicon with higher coverage are possible ways of improving our method. In our experiments, we used as neutral threshold the value 0.1 (i. e., the article with a score between -0.1 and 0.1 belongs to the neutral class). It would be interesting to perform tests to find out what value for the neutral threshold would result in better accuracy. Another good direction for future work is to take into account only terms near the keywords related to an article's topic to calculate the polarity of the article.

References

1. Pang, B., Lee, L., Vaithyanathan, S.: Thumbs up? Sentiment classification using machine learning techniques. In: Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP). (2002) 79–86
2. Melville, P., Gryc, W., Lawrence, R.D.: Sentiment analysis of blogs by combining lexical knowledge with text classification. In: KDD '09: Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining, New York, NY, USA, ACM (2009) 1275–1284

3. Kennedy, A., Inkpen, D.: Sentiment classification of movie reviews using contextual valence shifters. Computational Intelligence **22** (2006) 2006
4. Turney, P.D.: Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews. In: ACL '02: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, Morristown, NJ, USA, Association for Computational Linguistics (2002) 417–424
5. Gamon, M.: Sentiment classification on customer feedback data: noisy data, large feature vectors, and the role of linguistic analysis. In: In COLING. (2005) 841–847
6. Turney, P., Littman, M.: Measuring praise and criticism: Inference of semantic orientation from association. ACM Transactions on Information Systems **21** (2003) 315–346
7. Nigam, K., Hurst, M.: Towards a robust metric of polarity. In Shanahan, J.G., Qu, Y., Wiebe, J., eds.: Computing Attitude and Affect in Text: Theory and Applications. Volume 20 of The Information Retrieval Series. Springer-Verlag, Berlin/Heidelberg (2006) 265–279
8. Durant, K.T., Smith, M.D.: Predicting the political sentiment of web log posts using supervised machine learning techniques coupled with feature selection. In Nasraoui, O., Spiliopoulou, M., Srivastava, J., Mobasher, B., Masand, B.M., eds.: WEBKDD. Volume 4811 of Lecture Notes in Computer Science., Springer (2006) 187–206
9. Turney, P.D., Littman, M.L.: Unsupervised learning of semantic orientation from a hundred-billion-word corpus. CoRR **cs.LG/0212012** (2002)
10. Phan, X.H.: Jtextpro: A java-based text processing toolkit (2006) http://jtextpro.sourceforge.net/.
11. Esuli, A., Sebastiani, F.: SentiWordNet: A publicly available lexical resource for opinion mining. In: Proceedings of LREC-06, 5th Conference on Language Resources and Evaluation, Genova, IT (2006) 417–422
12. Taylor, A., Marcus, M., Santorini, B.: The penn treebank: An overview (2003)
13. Blitzer, J., Dredze, M., Pereira, F.: Biographies, bollywood, boomboxes and blenders: Domain adaptation for sentiment classification. In: In ACL. (2007) 187–205
14. Blitzer, J., Crammer, K., Kulesza, A., Pereira, O., Wortman, J.: Learning bounds for domain adaptation. In: In Advances in Neural Information Processing Systems. (2008)
15. Adomavicius, G., Tuzhilin, A.: Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. IEEE Trans. Knowl. Data Eng. **17**(6) (2005) 734–749
16. Thomas, M., Pang, B., Lee, L.: Get out the vote: Determining support or opposition from Congressional floor-debate transcripts. In: Proceedings of EMNLP. (2006) 327–335
17. Whitehead, M., Yaeger, L.: Building a general purpose cross-domain sentiment mining model. Computer Science and Information Engineering, World Congress on **4** (2009) 472–476
18. Schirru, R., Obradović, D., Baumann, S., Wortmann, P.: Domain-specific identification of topics and trends in the blogoshere. To appear in: Perner, P. (ed.) ICDM 2010. LNCS (LNAI), vol. 6171, Springer, Heidelberg (2010)