EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Bachelor's Thesis

Submitted in Partial Fulfillment of the Requirements
for the Degree of
Bachelor of Arts in Computational Linguistics

Comparing and Combining Rule Based and Machine Learning Techniques for Sentiment Analysis

Author:
Valentin Deyringer

Supervisor:
Ph.D. Christopher Culy

Seminar für Sprachwissenschaft
Eberhard-Karls-Universität Tübingen

August 2014

I hereby declare that this paper is the result of my own independent scholarly work. I have acknowledged all the other authors' ideas and referenced direct quotations from their work (in the form of books, articles, essays, dissertations, and on the internet). No material other than that listed has been used.

Tübingen, August 26, 2014

Valentin Deyringer

Contents

List of Figures **iv**

List of Tables **v**

1 Introduction **1**

2 Sentiment Analysis **1**

2.1 Motivation for Sentiment Analysis 2
2.2 Difficulties in Sentiment Analysis 3
2.3 Approaches to Sentiment Analysis 4
2.3.1 Support Vector Machines 5
2.3.2 Naïve Bayes Classification 5
2.3.3 Rule Based Approaches 6

3 The Experiment **7**

3.1 Resources 7
3.2 Implemented Algorithms 8
3.3 Results 9
3.4 Conclusion 12
3.5 Future Work 12

4 Online Sentiment Analysis Tool **13**

5 Acknowledgments **13**

References **14**

Abstract

This paper compares a rule based system and two machine learning techniques as approaches to sentiment analysis. The machine learning techniques used are Naïve Bayes classification and Support Vector Machines, both trained on a set of movie reviews, restaurant reviews and a combination of these two data sets. Afterwards combinations of the used classification methods are tested. It is found that the machine learning techniques yield better results than the rule based method. By combining the algorithms the misclassification rate can be reduced considerably. Furthermore a tool implementing the three algorithms to measure and visualize sentiment on the text level is presented.

List of Figures

2.1 Russell's Model of Emotional Affect 2
2.2 Scatterplot Featuring a Linear Support Vector Machine's Decision Boundary 6

List of Tables

3.1 Accuracies of Naïve Bayes Classifiers on Different Test Sets 10
3.2 Correct Classifications of Naïve Bayes Classifier Trained on Yelp Training Data Set 10
3.3 Accuracies of Support Vector Machines on Different Test Sets . . 11
3.4 Accuracy of Rule Based Approach on Different Test Sets 11
3.5 Accuracies and Misclassification of Combined Classifications for Combined Test Data Set 11

1 Introduction

Natural Language Processing (NLP) is a discipline concerned with the automation of linguistic processes by means of algorithmic solutions. The ultimate goal of NLP is to make machines understand human language. For a more precise introduction to NLP see (Liddy, 2001).
The knowledge gained by NLP is not only of interest for linguists or cognitive scientists but also can provide useful information in other areas.
An important branch of NLP is text classification which aims to assign categories to a text. These categories can be of various types. For instance one could identify the gender of an author given the author's writings (Argamon, Koppel, Fine, & Shimoni, 2003).
Another text classification task which has drawn a lot of interest over the last few years is sentiment analysis. Sentiment analysis aims to extract the subjective information a text conveys. The following section will give a more precise description of sentiment analysis, its motivation and the difficulties it faces.
This work will present different approaches to sentiment analysis, analyse some of these approaches and evaluate the results. Additionally combined approaches are tested. The results of these evaluations helped towards developing a tool which is then presented. This tool offers an interface to perform sentiment analysis for texts based on the algorithms implemented for the experiment. A main feature of the tool is the visualization of the outputs.

2 Sentiment Analysis

Sentiment analysis or opinion mining is the task of extracting the subjective information of a text. This task can be performed in a general fashion by extracting the overall sentiment of a document or more specifically by classifying several parts of a document. There is also a difference between general sentiment of a document and topic dependent sentiment, i.e. the authors attitude towards a specific topic.
Sentiment values can be represented on different scales. Some work just focuses on the two levels of *positive* and *negative* sentiment, or on an extended scale with the additional levels of *neutral* and/or *mixed* sentiment. Other systems offer more fine grained scales. For example a scale from -10 to $+10$ with negative values representing negative sentiment and positive values representing positive sentiment could be used. It is also possible to represent sentiment on a more dimensional

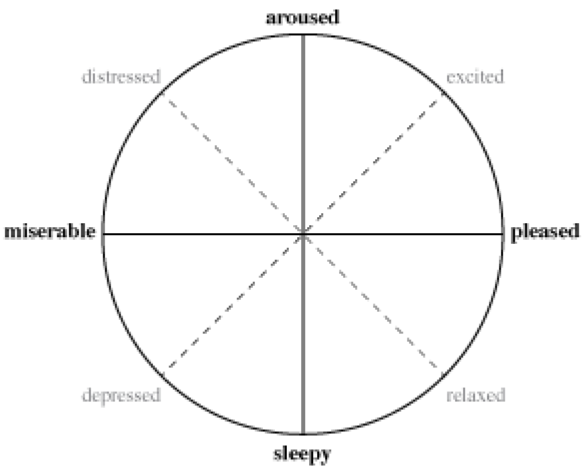

Figure 2.1: Russell's Model of Emotional Affect (*Russell's Model of Emotional Affect*, 2013)

scale. *Tweet Viz*[1] for instance places documents on a two-dimensional *emotional plane*. Similar to the model presented by (Russell, 1980) shown in Figure 2.1, this plane is defined by the axis of activeness ranging from subdued to active and the axis of pleasure ranging from highly unpleasant to highly pleasant. Emotions could be represented in even more dimensions. The *ANEW*[2] dictionary presented in (Bradley & Lang, 1999) for instance lists words with their ratings for the three dimensions of valence, arousal and dominance.
The task of sentiment analysis has been approached with different methods which are machine learning based or rule based. The most common approaches are explained in Section 2.3

2.1 Motivation for Sentiment Analysis

There are numerous free and commercial systems implementing sentiment analysis algorithms. These systems serve different purposes. The most popular field is investigating the reputation of companies and their products. There are several companies providing social media monitoring systems which also display sentiment measures.
Another application area is the financial market. (Bollen, Mao, & Zeng, 2011) examined the correlation between the sentiment conveyed by huge amounts of data from the microblogging platform twitter and the DJIA[3].
(Kramer, Guillory, & Hancock, 2014) examined the phenomenon of emotional

[1] http://www.csc.ncsu.edu/faculty/healey/tweet_viz/
[2] Affective Norms for English Words
[3] Dow Jones Industrial Average

contagion via text in the social network facebook in a recent study.

2.2 Difficulties in Sentiment Analysis

Sentiment analysis is not a trivial task. (Da Silva Cardoso, 2013), based on (S. Mukherjee & Bhattacharyya, 2013) lists the following difficulties:

- Sarcasm, irony and implicit sentiments
  The words *only* and *lovely* in *"My computer crashed only ten times this week. What a lovely machine!"* are not meant literally but rather express an opposite negative sentiment.

- Domain dependency
  In general the sentence *"Did he read the book?"* does not convey sentiment. But when this statement is made about a director of a movie in a movie review it is negative. (Pang, Lee, & Vaithyanathan, 2002)

- "Thwarted expectations" (Pang et al., 2002)
  A previous statement might be reversed afterwards like in the following statement *"Based on the book and the cast this should be the best movie ever. But it isn't!"*

- Pragmatics
  Special characteristics of a text which the character representation does not resemble may play a role for the sentiment. For example words may be underlined or written in italic for emphasis. This also has an effect on the sentiment these words convey.

- World knowledge
  "Referring to commonly known subjects, as: You must be Einstein! suggests that a person is smart and should be detected as positive, although Einstein would not generally carry a positive connotation."(Da Silva Cardoso, 2013)

- Subjectivity detection
  The sentence *"I hate love stories."* (A. Mukherjee & Liu, 2012) combines the negative word *hate* and the positive word *love*. A human reader resolves this consecutiveness without problems. A machine in contrast may struggle to resolve such cases.

- Entity Identification
  The sentence *"I really like my new Mercedes, it's so much more comfortable*

than my old rusty Ford." is a positive statement with *Mercedes* as target word. The statement is negative when *Ford* is targeted. Grammatical analysis is needed to resolve which word the sentiment bearing words are targeted at.

- Anaphora
  It is a very challenging task for machines to interpret which entity a pronoun refers to like *he* in *"Lenoardo DiCaprio is a way better actor than Matt Damon. He definitely deserves an academy award."*. This is especially important for sentiment analysis with a target word.

- Negation
  A generally positive word might be negated and thus convey negative sentiment and vice versa. *"There was not one good scene in the whole movie!"* The algorithms implemented for this work contain a basic negation handling procedure.

2.3 Approaches to Sentiment Analysis

For classification in general and sentiment analysis in particular, the state of the art techniques are based on machine learning algorithms. Machine learning algorithms tempt to detect patterns in a given training data set. The recognized structures then are used to process unseen data. In the case of classification this process aims to assign a class to a new data point.
There are supervised and unsupervised machine learning algorithms. Supervised algorithms are given labelled data and the algorithms try to find the best features describing the different classes.
For unsupervised algorithms the training data is not labelled. The algorithms cluster similar data points and separate the different classes this way.
There are also intermediate machine learning algorithms. So called semi-supervised algorithms combine two or more different algorithms which start with labelled examples, and then learn from interacting with the other algorithms. Active learning techniques are basically supervised methods but ask a user to confirm the outputs for difficult cases and learn and improve its assumptions based on the new inputs.
In the following we will discuss the machine learning techniques which find the most application for sentiment analysis in detail. These are the supervised methods of Naïve Bayes classification and Support Vector Machines. Other machine

learning algorithms like Maximum Entropy classification are not taken into account for this work.
Both techniques are examined more precisely in the experiment in Section 3 and implemented in the tool presented in Section 4. Additionally rule based approaches are presented out of which one also is part of the experiment and the implemented tool.

2.3.1 Support Vector Machines

The machine learning approach of Support Vector Machines has been implemented in several systems and is found to be performing well for classifying texts for their sentiment (Syamlal & Bruins, 2007; Pang et al., 2002).
Support Vector Machines represent the data points as feature vectors in a feature space. The algorithm then fits a hyperplane separating the data such that the distance of the hyperplane and the data is maximal. Figure 2.2 shows an example of such a hyperplane separating data points linearly. We can see that the data is not linearly separable. To still fit the hyperplane correctly it needs to be bent. This can be achieved by applying the so called *kernel trick*. This method is based on a transformation of the space into a higher dimension such that the data becomes linearly separable. The space then is transformed back into the original dimension. The resulting hyperplane then is not linear any more but correctly separates the data into the different classes. New data points are classified by computing their feature vectors and assigning the class according to which side of the hyperplane the data point is found in the feature space. Additional slack variables make Support Vector Machines more flexible.
Basically Support Vector Machines allow only two classes. It is still possible to create Support Vector Machines which can handle more classes. This can be achieved by calculating hyperplanes hierarchically per class while treating all other classes as one opposite class.

2.3.2 Naïve Bayes Classification

Also the method of Naïve Bayes Classification has been implemented successfully for sentiment analysis (Narayanan, Arora, & Bhatia, 2013).
A Naïve Bayes Classifier is a maximum likelihood classifier based on the Bayes Theorem. The probabilities of each feature for all classes are computed on the training data set. Given the features of an unseen data point the probability $p(A)$ of that data point belonging to class A is computed as

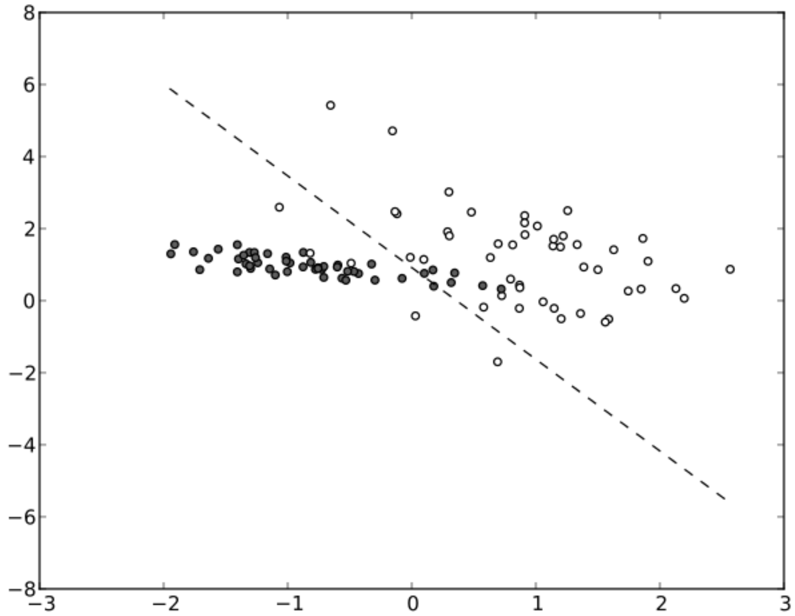

Figure 2.2: Scatterplot Featuring a Linear Support Vector Machine's Decision Boundary (Dashed Line) (Qwertyus, 2014)

$$p(A) = \sum_{i=1}^{k} log(p_A(f_i)) \tag{1}$$

with f_i as a feature of the new data point. The frequency of f_i occurring in class A $p_A(f_i)$ is computed as

$$p_A(f_i) = \frac{l_A(f_i) + 1}{n \times m} \tag{2}$$

with $l_A(f_i)$ being the count of the feature for class A in the training data set, n as the number of classes and m being the number of all features observed in the training data set. By adding 1 to the feature count, features of the new data point not observed in the training data are accounted for. This step is called laplacian smoothing.
The class that the highest probability was computed for is assigned to the new data point.

2.3.3 Rule Based Approaches

In contrast to machine learning techniques, it is also possible to classify a text with predefined rules. These rules consist of an antecedent and a sentiment value

as consequent. The antecedent can be a word or a more complex construct like a combination of words that need to appear in a certain distance.
A set of rules can be defined as a lexicon of positive words and a lexicon of negative words. An example for such a sentiment lexicon is one that is used for the general inquirer consisting of 1637 words classified as positive and 2007 words classified as negative (Stone, Dunphy, & Smith, 1966). Another lexicon is used for the commercial system LIWC[4] (Pennebaker, Chung, Ireland, Gonzales, & Booth, 2007).
Rules can be extracted from pre classified texts or by estimating a resulting sentiment value for the antecedents in another way. (Prabowo & Thelwall, 2009) for example implemented an algorithm which computed sentiment values of worfieds based on hit counts of searches performed on google. The used queries basically consisted of the word to compute a value for and a predefined word with strong sentiment.
It is also possible to construct a lexicon with help of semantic nets like *WordNet* (Miller, Beckwith, Fellbaum, Gross, & Miller, 1990). (Hu & Liu, 2004) and (Kim & Hovy, 2004) Implemented algorithms which started with a seed set of a few sentiment bearing words and collecting all words that could be associated with these by traversing *WordNet*.
For classification the text to classify is scanned for the rules' antecedents. The sentiment value of the text than can be computed with all the consequences of the rules that applied.

3 The Experiment

For this work an experiment was carried out which compared Support Vector Machines, Naïve Bayes Classification and a rule based approach. Subsequently the different algorithms were combined to minimize the misclassification rate.
All computations were performed on a Laptop running an Intel Core i5 at 1.7GHz.

3.1 Resources

For setting up supervised machine learning algorithms like Naïve Bayes Classification and Support Vector Machines, a training data set is needed.
A further test data set is required for feature selection and to estimate the quality of the classification methods. Test data sets are also meaningful for non machine learning techniques.

[4]Linguistic Inquiry and Word Count

One of the used data sets is a set of movie reviews from the website www.imdb.com as introduced in (Maas et al., 2011). This dataset contains 50000 reviews evenly split into a training set of 25000 reviews and a test set of 25000 reviews. The overall distribution of labels is balanced such that there are 25000 positive reviews and 25000 negative reviews, 12500 for training and 12500 for testing each.
The second data set used is the Yelp academic data set[5]. This set contains 330071 restaurant reviews with ratings from one to five stars. Reviews with five stars were assumed to be positive, reviews with one star were assumed to be negative. It is important to note that the correlation between star ratings and sentiment is not a perfect agreement. Five star ratings may contain negative features as well as one star ratings may contain positive features. However, by manually checking a small set of ten randomly selected reviews for each polarity the assumptions made could be confirmed. One negative review which was an edited review constituted an exception. The author cited his previous positive text and then added his new negative text. This phenomenon is an instance of thwarted expressions as presented in Section 2.2 and is not considered further in this work. The original data set contained 95807 reviews rated with five stars and 26383 reviews rated with one star. For comparison reasons this data set was scaled down to 50000 texts, split into subcategories identical to the IMDb data set. The rest of 72190 reviews with one or five stars were discarded. The ratio between positive and negative reviews in the original data was not kept. A classifier trained on an unbalanced data set would greatly weight features of the smaller class which is not desirable in the case of sentiment analysis. The leftover reviews could have been added to the test set. But to keep the size of the different test data sets comparable this was not done. Reviews rated with 2, 3 or 4 stars were not taken into account for this work.
A third data set of 50000 texts was composed from 25000 texts of the IMDb data set and 25000 texts of the Yelp data set, 12500 positive and 12500 negative texts each. This third data set also contains 25000 texts for training and 25000 texts for testing, evenly split in positive and negative texts.

3.2 Implemented Algorithms

The rule based algorithm implemented for the experiment is based on the sentiment lexicon of (Stone et al., 1966) as explained in Section 2.3.3. The input text needs to be split into tokens and these tokens need to be lemmatized to match the dictionaries entries. Additionally a basic negation handling tries to detect in-

[5] The Yelp academic data set is available at www.yelp.com/academic_dataset

verting words. The sentiment value is then computed on the counts of sentiment bearing words found in the text. If no such word is found in the text or the counts of positive and negative words are equal, no sentiment value is assigned. The two sets of words have an overlap of 15 words[6] which are considered as conveying mixed sentiment and do not have an influence on the counts.
Based on the implementation of Naïve Bayes Classification presented in (Narayanan et al., 2013) the features to use for the machine learning approaches were examined. A slight change was made by changing the tokenization step to the built-in tokenization method of NLTK. This way the basic implementation achieved an accuracy of 87.5% with a computation time of about 3.5 minutes. Lemmatizing the tokens decreased that accuracy to 87.2% while training time was more than threefold the initial training time. The time consumption is attributable to the process of Part of Speech tagging which is needed for correct lemmatization. Extending the feature sets by token fourgrams also resulted in less accurate classification with 87.4% accuracy while training time increased by about 50 seconds. Based on these results the features taken into account for the experiment were token uni-, bi- and trigrams with basic negation handling. As the findings in (Narayanan et al., 2013) propose, the features extracted from the training sets were restricted to the top 32000 features based on mutual information. This restriction also keeps training time of the classifiers reasonable.
Since the implementations differ, other features and feature selection methods could achieve better results. However, the features were kept the same for comparison reasons.
The Naïve Bayes Classifier used is the built-in algorithm provided by NLTK. The Support Vector Machine used is an implementation offered by sklearn in the scipy toolkit. All default values of slack variables were retained.
After reading in the data and performing all preprocessing steps, the training of the Naïve Bayes Classifiers took 26.67 seconds in average and training the Support Vector Machines took 16.92 seconds in average.

3.3 Results

The different classifiers were all tested on the three test data sets.
Table 3.1 shows the accuracy of the different Naïve Bayes Classifiers. For the Naïve Bayes Classifiers we can see that the classifier trained on the IMDb training data set yielded the best results when averaged over the test data sets. The lowest

[6]Words appearing in both word lists: *arrest*, *board*, *deal*, *even*, *fine*, *fun*, *hand*, *help*, *hit*, *laugh*, *make*, *matter*, *order*, *particular*, *pass*

	IMDb Data	Yelp Data	Combined Data	Average
Trained on IMDb	**87.676 %**	84.136 %	**85.976 %**	**85.929** %
Trained on Yelp	61.104 %	85.384 %	73.272 %	73.252 %
Trained on Comb.	81.432 %	**87.608 %**	84.424 %	84.488 %

Table 3.1: Accuracies of Naïve Bayes Classifiers on Different Test Sets

	IMDb Data	Yelp Data	Comb. Data
correct positive	23.144 %	72.144 %	47.536 %
correct negative	99.064 %	98.624 %	99.008 %

Table 3.2: Correct Classifications of Naïve Bayes Classifier Trained on Yelp Training Data Set

accuracy is obtained when applying the classifier trained on the Yelp data set to the IMDb test set.

When examining the results closer, the source for this low accuracy is found. The balance of correct classifications for the two classes made by the Naïve Bayes Classifier trained on the Yelp training data set is shown in Table 3.2. This classifier has a strong tendency towards assigning the texts to the negative class. Correct assignments for the positive class are rare and the accuracy for assigning the class positive to a post drops as low as 23% for the IMDb test data set.

The Naïve Bayes Classifier trained on the combined training data set showed similar but less stark behaviour. This is due to the combined training data set including a subset of the Yelp training data set.

Furthermore the Naïve Bayes Classifiers are struggling to assign the positive class correctly when tested on the Yelp test data set.

In Table 3.3 the accuracies of the different Support Vector Machines are listed.

The Support Vector Machine trained on the combined data set has the highest average accuracy. Only the accuracy of the classifier trained on the Yelp data set applied to the IMDb test set is below 80%. Even that accuracy is almost 77% which is fairly good. Some of the accuracies even exceed 90%. Compared to the Naïve Bayes Classifiers only the Support Vector Machine trained on the IMDb data set is slightly inferior to its counterpart. We can also see that the Support Vector Machines performing best on a test data set are the ones trained on the corresponding training data set. This is an indicator of the Support Vector Machines overfitting for the domain they are trained on.

The results of the rule based approach are shown in Table 3.4.

Compared to the results of the machine learning techniques the results of the rule based algorithm are relatively poor. Compared to a baseline classification

	IMDb Data	Yelp Data	Combined Data	Average
Trained on IMDb	**85.812 %**	80.420 %	83.132 %	83.121 %
Trained on Yelp	76.976 %	**92.152 %**	84.344 %	84.491 %
Trained on Comb.	84.296 %	91.276 %	**87.696 %**	**87.756** %

Table 3.3: Accuracies of Support Vector Machines on Different Test Sets

IMDb Data	Yelp Data	Combined Data	Average
63.676 %	64.740 %	63.864 %	64.093 %

Table 3.4: Accuracy of Rule Based Approach on Different Test Sets

assigning the same class to all texts in the training set (50% accuracy) these results are still good when taking into account the very easy implementation.
Note that the results achieved on the combined training data set are close to the mean of the results on the other two data sets. This is not surprising since the combined data set consists of evenly sized subsets of the other two data sets
Table 3.5 shows the accuracies and misclassification rate of combined approaches tested on the combined test data set. For this computation the Naïve Bayes Classifier trained on the IMDb training data set and the Support Vector Machine trained on the combined training data set were chosen since they yielded the best averaged results in previous computations.
By combining the different approaches we can achieve a great improvement of precision. We can see that the misclassification rate for the combined data set dropped considerable from between 36.2% for rule based classification and 12.3% for Support Vector Machine classification to maximally 5.22% with a combination of these two techniques. However, the recall is also decreased to a minimal value of 52.6% for the combination of all three algorithms.

	Correct Classifications	Misclassifications
NB + Rules	56.148 %	4.832 %
SVM + Rules	58.080 %	5.248 %
NB + SVM	78.776 %	5.104 %
NB + SVM + Rules	52.596 %	2.308 %

Table 3.5: Accuracies and Misclassification of Combined Classifications for Combined Test Data Set

3.4 Conclusion

Based on the achieved results the machine learning techniques perform better than the rule based approach. The Support Vector Machines then are slightly better than Naïve Bayes classification in most cases.
Combining different algorithms can help increasing the precision greatly. But this improvement comes at the cost of a decreased recall.

3.5 Future Work

To improve the obtained results the sentiment lexicon used for the rule based approach could be extended and adjusted. The rules could also be extend to taking into account more words which lead to a sentiment value when appearing in a specified proximity. A way of adding single words to the lexicon is implemented in the tool presented in Section 4.
While the effect of negation is taken into account, this work does not consider the effect of valence shifters like intensifiers and diminishers. (Kennedy & Inkpen, 2006) show that these can also have an impact on the obtained results. In addition the implemented negation handling is very basic and could be improved further.
Also the data sets used for training could be improved by manually checking their grading based on the star ratings. Because of the large size of these data sets this might be very time and cost intensive and therefore not feasible. Especially the Yelp data seems to pose some difficulties which are beyond the scope of this work. These difficulties could come from an incorrect mapping of star ratings and sentiment values as described in Section 3.1. Since all slack variables were retained an adjustment for these may preventof the Support Vector Machine algorithm could be adjusted which may prevent overfitting and improve the results further.
Since the used data of online reviews usually also contains emoticons (*:-)*, *:-(*, *;)*, ...) the built-in tokenization of NLTK which is used fails. As improvement, the tokenization method of (O'Connor, Krieger, & Ahn, 2010) called *twokenize*[7] could be used. However, when taking into account token bigrams and trigrams, emoticons split into separate tokens would be considered as one again and make this improvement unnecessary.
(Wang & Manning, 2012) reports that Naïve Bayes classification performs better on short texts and Support Vector Machine performs better on long texts. This findings could also be measured more precisely and improve the results. Other aspects of a text could enhance the results of the different approaches even further.

[7] available at https://github.com/brendano/tweetmotif

The feature selection performed before training of the machine learning classifiers is a baseline as suggested by (Narayanan et al., 2013). The features could also be selected by the TF-IDF measure or with the methods presented in (O'Keefe & Koprinska, 2009).

4 Online Sentiment Analysis Tool

5 Acknowledgments

References

Argamon, S., Koppel, M., Fine, J., & Shimoni, A. R. (2003). Gender, genre, and writing style in formal written texts. *To appear in Text*, *23*, 3.

Bollen, J., Mao, H., & Zeng, X. (2011). Twitter mood predicts the stock market. *Journal of Computational Science*, *2*(1), 1–8.

Bradley, M. M., & Lang, P. J. (1999). *Affective norms for english words (anew): Instruction manual and affective ratings.*

Da Silva Cardoso, H. K. (2013). Sentiment analysis of correspondence corpora.

Hu, M., & Liu, B. (2004). Mining and summarizing customer reviews. In *Proceedings of the tenth acm sigkdd international conference on knowledge discovery and data mining* (pp. 168–177).

Kennedy, A., & Inkpen, D. (2006). Sentiment classification of movie reviews using contextual valence shifters. *Computational Intelligence*, *22*(2), 110–125.

Kim, S.-M., & Hovy, E. (2004). Determining the sentiment of opinions. In *Proceedings of the 20th international conference on computational linguistics* (p. 1367).

Kramer, A. D., Guillory, J. E., & Hancock, J. T. (2014). Experimental evidence of massive-scale emotional contagion through social networks. *Proceedings of the National Academy of Sciences*, 201320040.

Liddy, E. D. (2001). Natural language processing.

Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., & Potts, C. (2011, June). Learning word vectors for sentiment analysis. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies* (pp. 142–150). Portland, Oregon, USA: Association for Computational Linguistics. Available from `http://www.aclweb.org/anthology/P11-1015`

Miller, G. A., Beckwith, R., Fellbaum, C., Gross, D., & Miller, K. J. (1990). Introduction to wordnet: An on-line lexical database*. *International journal of lexicography*, *3*(4), 235–244.

Mukherjee, A., & Liu, B. (2012). Modeling review comments. In *Proceedings of the 50th annual meeting of the association for computational linguistics: Long papers-volume 1* (pp. 320–329).

Mukherjee, S., & Bhattacharyya, P. (2013). Sentiment analysis: A literature survey. *arXiv preprint arXiv:1304.4520*.

Narayanan, V., Arora, I., & Bhatia, A. (2013). Fast and accurate sentiment classification using an enhanced naive bayes model. In *Intelligent data engineering and automated learning–ideal 2013* (pp. 194–201). Springer.

O'Connor, B., Krieger, M., & Ahn, D. (2010). Tweetmotif: Exploratory search and topic summarization for twitter. In W. W. Cohen, S. Gosling, W. W. Cohen, & S. Gosling (Eds.), *Icwsm.* The AAAI Press.

O'Keefe, T., & Koprinska, I. (2009). Feature selection and weighting methods in sentiment analysis. *ADCS 2009*, 67.

Pang, B., Lee, L., & Vaithyanathan, S. (2002). Thumbs up?: sentiment classification using machine learning techniques. In *Proceedings of the acl-02 conference on empirical methods in natural language processing-volume 10* (pp. 79–86).

Pennebaker, J. W., Chung, C. K., Ireland, M., Gonzales, A., & Booth, R. J. (2007). The development and psychometric properties of liwc2007. *Austin, TX, LIWC. Net*.

Prabowo, R., & Thelwall, M. (2009). Sentiment analysis: A combined approach. *Journal of Informetrics*, *3*(2), 143–157.

Qwertyus, W. U. (2014). *Linear-svm-scatterplot.* Available from `http://upload.wikimedia.org/wikipedia/commons/4/46/Linear-svm-scatterplot.svg`

Russell, J. A. (1980). A circumplex model of affect. *Journal of personality and social psychology*, *39*(6), 1161.

Russell's model of emotional affect. (2013). Available from `http://www.csc.ncsu.edu/faculty/healey/tweet_viz/figs/circumplex.png`

Stone, P. J., Dunphy, D. C., & Smith, M. S. (1966). The general inquirer: A computer approach to content analysis.

Syamlal, R., & Bruins, J. (2007). Sentiment classification using a svm.

Wang, S., & Manning, C. D. (2012). Baselines and bigrams: Simple, good sentiment and topic classification. In *Proceedings of the 50th annual meeting of the association for computational linguistics: Short papers-volume 2* (pp. 90–94).