A Statistical Approach to Star Rating Classification of Sentiment

Alexander Hogenboom, Ferry Boon, and Flavius Frasincar

Abstract Automated analysis of the ever-increasing amount of reviews available through the Web can enable businesses to identify why people like or dislike (aspects of) products or brands, yet to this end, a reliable indication of the intended sentiment of reviews is of crucial importance. This sentiment is typically quantified in universal star ratings, which are not always available. We propose and compare the performance of several statistical methods of automatically classifying star ratings of reviews represented by means of a binary vector representation, with features signaling the presence of sentiment-carrying words. A nearest neighbor classifier maximizes recall, whereas a naïve Bayes classifier excels in terms of precision, accuracy, and the root mean squared error of the assigned number of stars.

Key words: Sentiment analysis, star ratings, nearest neighbor, naïve Bayes

1 Introduction

The Web as it exists today encompasses a vast and ever-increasing amount of user-generated content. Popular Web sites like Twitter, Blogger, or Epinions enable anyone to write and publish short messages, blog posts, or reviews about anything at any time. Today's typical Web user exhibits a hunger for and reliance upon on-line advice and recommendations, yet in the wealth of user-generated content, explicit information on user opinions is often hard to find, confusing, or overwhelming [11]. Nevertheless, user-generated content does contain traces of people's sentiment. As recent estimates indicate that twenty percent of all tweets [6] and one third of all blog posts [8] discuss products or brands, automated information monitoring tools for consumer sentiment are crucial for today's businesses.

Alexander Hogenboom · Ferry Boon · Flavius Frasincar
Erasmus University Rotterdam, P.O. Box 1738, NL-3000 DR, Rotterdam, the Netherlands, e-mail: hogenboom@ese.eur.nl, ferry.boon@gmail.com, frasincar@ese.eur.nl

For such information systems, reviews form an important source of information for, e.g., marketing and reputation management. In reviews, users describe their experiences with a particular brand or product, while implicitly or explicitly expressing what they do or do not like about the subject of their respective reviews. The overall verdict of a review can typically be classified by means of universal star ratings, where the number of stars reflects the extent to which a reviewer intends to convey positive sentiment with respect to the review's subject. Such star classes, typically five, are defined on an ordinal scale, e.g., a piece of text that is assigned five stars is considered to be more positive than a four-star piece of text.

Star ratings can enable the extraction of valuable information from the multitude of available reviews, as they can facilitate analyses of, e.g., which aspects of an arbitrary product are mentioned in what context in reviews associated with particular ratings. Sentiment analysis techniques can be used to this end. Some of such techniques focus on identifying the subjectivity or objectivity of a text, whereas other techniques aim to determine the polarity of natural language text.

Typical sentiment analysis approaches involve scanning a text for cues signaling subjectivity or polarity, e.g., words, parts of words, or other (latent) features of natural language text, typically in statistics-based machine learning approaches. The use of sentiment lexicons – lists of words and their associated sentiment, possibly differentiated by Part-of-Speech (POS) and/or meaning [1] – has gained attention in recent research endeavors [2, 3]. Such lexicon-based methods have been shown to have a more robust performance across domains and texts than pure machine learning approaches [14]. Additionally, lexicon-based methods allow for intuitive ways of incorporating deep linguistic analysis into the sentiment analysis, for instance by accounting for structural or semantic aspects of text, but this comes at a cost of significant decreases in processing speed with respect to statistical approaches [2].

In order to be able to (semi-)automatically analyze user-generated content for clues as to, e.g., why people like or dislike (aspects of) products or brands, or how different aspects of products contribute to the overall user experience, a reliable indication of intended sentiment associated with this content is of crucial importance. Some Web sites offer users the possibility to assign scores to their reviews in order to express their intended sentiment, but such scores are not always available. For instance, opinionated blog posts or tweets are not typically assigned scores by their respective authors in order to signal their intended sentiment. Therefore, a major challenge is to automatically determine the star rating associated with reviews based on cues in the actual natural language content.

In this light, we propose and compare several statistical methods for classifying the star rating of reviews. In our current endeavors, we aim to contribute to combining the accuracy and processing speed benefits of statistics-based sentiment analysis approaches with the robustness of lexicon-based approaches.

The remainder of this paper is structured as follows. First, we discuss related work on sentiment analysis in Sect. 2. Then, we propose several statistics-based approaches to star rating classification of the sentiment associated with reviews in Sect. 3. An evaluation of our methods is presented in Sect. 4. Last, we conclude and propose directions for future work in Sect. 5.

2 Sentiment Analysis

The research area of sentiment analysis is related to natural language processing, computational linguistics, and text mining. The main goal of sentiment analysis is the extraction of subjective information from natural language text. Existing work focuses on several specific tasks. Some work aims to distinguish subjective text segments from objective ones or to identify the degree of subjectivity of text [17]. Other work is focused on determining the overall polarity of words, sentences, text segments, or documents [11]. This is typically treated as a binary classification problem, i.e., text is classified as either positive or negative, yet some research focuses on ternary classification by introducing a third class of neutral documents. Other work focuses on determining the degree of positivity or negativity of text.

In general, there are two main types of approaches to sentiment classification tasks. On the one hand, some approaches exploit (generic) sentiment lexicons when determining the subjectivity or polarity of natural language text. On the other hand, many state-of-the-art approaches rely on statistics-based machine learning techniques for sentiment analysis.

Lexicon-based approaches take into account the semantic orientation of individual words by matching words in a text with a list of words with their associated sentiment, possibly differentiated by POS and/or meaning. The overall semantic orientation of a text is then determined by aggregating (e.g., summing) the word scores, possibly while taking into account other aspects of content as well, e.g., negation [3, 5], intensification [13], or rhetorical roles of text segments [2, 4]. Lexicon-based approaches enable deep, yet computationally intensive linguistic analysis to be incorporated into the process of analyzing sentiment in natural language text [2] and have been shown to have a robust performance across domains and texts [14].

On the other hand, machine learning approaches have been shown to have great potential with respect to sentiment classification accuracy in specific domains for which they have been trained [14]. In such approaches, text is typically represented as a vector, which can be used to model the text as a bag-of-words, i.e., an unordered collection of words occurring in a document. Here, a binary representation of text, indicating the presence or absence of specific words [10] has been shown to be more effective than a frequency-based vector representation text [12]. Vectors may also contain features other than words, e.g., parts of words, word groups, or features representing other aspects of content such as semantic distinctions between words [16]. Features represented in vectors may be weighted as well [9].

Machine learning approaches have an attractive advantage over lexicon-based approaches in that they tend to perform better in terms of classification accuracy [14]. Additionally, lexicon-based methods tend to sacrifice computational efficiency when naturally incorporating deep linguistic analysis into the sentiment analysis process [2]. These properties render statistics-based machine learning techniques attractive approaches to sentiment analysis tasks. However, lexicon-based methods tend to be more robust across domains and texts [14]. Therefore, a statistics-based method in which sentiment lexicons are exploited as well appears to be a viable approach to our targeted multi-class sentiment analysis problem.

3 Star Rating Classification

In this paper, we aim to automatically determine the star rating of reviews by means of analyzing the sentiment conveyed by these pieces of natural language text. Rather than targeting a binary or ternary sentiment classification problem, we aim to distinguish five sentiment classes, i.e., one star, two stars, etcetera. These stars represent sentiment classifications ranging from very negative (one star), to neutral (three stars), and very positive (five stars).

As we hypothesize that the boundaries between classes may not be very clear-cut because of the different degrees of positivity and negativity represented by our star ratings, we assume that a statistics-based machine learning approach would be a better fit than unsupervised lexicon-based approaches for the problem we target in our current endeavors. Nevertheless, the robustness across domains and texts typically exhibited by lexicon-based approaches is an appealing feature. In this light, we propose to make a first step towards combining the classification accuracy and processing speed benefits of statistics-based sentiment analysis approaches with the robustness of lexicon-based approaches by means of linking vector representations of our texts to a sentiment lexicon.

In order to be able to apply statistical analyses on our data, we need a proper representation of our texts. We propose a novel *bag-of-sentiwords* representation, i.e., a vector with features representing the presence of sentiment-carrying words, retrieved from a sentiment lexicon. We include only sentiment-carrying words in our vector, as we assume these words to play a major, if not crucial role in conveying the overall sentiment of a text, as opinionated texts significantly differ from non-opinionated texts in terms of occurrences of subjective words [15]. We propose to use a binary representation, as we hypothesize that the sentiment conveyed by a text is not so much in the number of times a single word occurs in a text, but rather in the (number of) distinct words with a similar semantic orientation. Moreover, research has shown that such a binary representation is more effective for sentiment analysis purposes than a frequency-based vector representation of natural language text [12].

Statistical analyses and machine learning algorithms can be applied to the vector representations of text thus obtained in order to identify similarities between texts and to exploit these, such that the correct sentiment classification of a text can be identified. In this work, we consider two types of classifiers, both of which assume the availability of a set of training data, labeled with their corresponding sentiment classification, and a set of test data for which the sentiment needs to be classified based on the model built from the training data. The first type of classifier we consider is a nearest neighbor classifier. Additionally, we consider to use a naïve Bayes classifier for determining the star rating associated with a text.

In our nearest neighbor classifier, we compare an arbitrary unlabeled text with vector representations of each of our considered classes and subsequently assign to the text the label of the class with which the similarity is the highest. These vector representations of classes are typically representative documents or they represent the typical characteristics of documents in their respective classes. The similarity between two (vector representations of) documents can be measured in several ways.

First, we consider to compute a Jaccard similarity coefficient, by defining the similarity $s_{\text{jac}}(d_i, d_j)$ between documents d_i and d_j as the size – i.e., the number of ones in the vector representation – of the intersection of d_i and d_j in terms of the size of the union of these documents, i.e.,

$$s_{\text{jac}}(d_i, d_j) = \frac{\left|d_i \cap d_j\right|}{\left|d_i \cup d_j\right|}. \tag{1}$$

Alternatively, the similarity between two vector representations of natural language text could be computed by means of the cosine similarity $s_{\cos}(d_i, d_j)$ of document d_i to document d_j, i.e.,

$$s_{\cos}(d_i, d_j) = \frac{\sum_{f=1}^{n} d_{i_f} d_{j_f}}{\sqrt{\sum_{f=1}^{n} \left(d_{i_f}\right)^2} \sqrt{\sum_{f=1}^{n} \left(d_{j_f}\right)^2}}, \tag{2}$$

with d_{i_f} and d_{j_f} representing feature f out of n features for documents d_i and d_j, respectively.

Another design issue lies in the definition of a class, i.e., the determination of which vector representation(s) an unlabeled document should be compared with in order to determine its class. In our current endeavors, we consider three types of vector representations of a class.

The first vector representation of a class we consider is a centroid representation, where a class is represented by the document with the highest similarity to all other documents in its class. When using this representation, an unlabeled document is assigned the class of the centroid that is most similar to this document.

Second, we consider representing each class by means of all its associated documents. This implies that a new document can be classified by computing its similarity to each document in the training set and subsequently classifying it into the class associated with the highest similarity, averaged over its constituting documents.

Last, we consider to represent each class by merging all documents in each respective class into one vector representation per class. In this merger, a new vector is constructed for an arbitrary class by taking the union of all vectors this class is constituted by. When using this representation, an unlabeled text can be classified into the class of which the merged vector has the highest similarity to the vector representation of the unlabeled text.

As an alternative to our considered nearest neighbor methods, we consider a naïve Bayes classifier. In this classifier, a document d_i is assigned a class c_k, for which the probability $P(c_k|d_i)$ is maximized. This probability is defined as the product of the prior probability $P(c_k)$ of class c_k to occur – which can be estimated from the training data – and the probability $P(w_t|c_j)$ of each of its m distinct words w_t to occur in a document of class c_k, i.e.,

$$P(c_k|d_i) = P(c_k) \prod_{t=1}^{m} P(w_t|c_j). \tag{3}$$

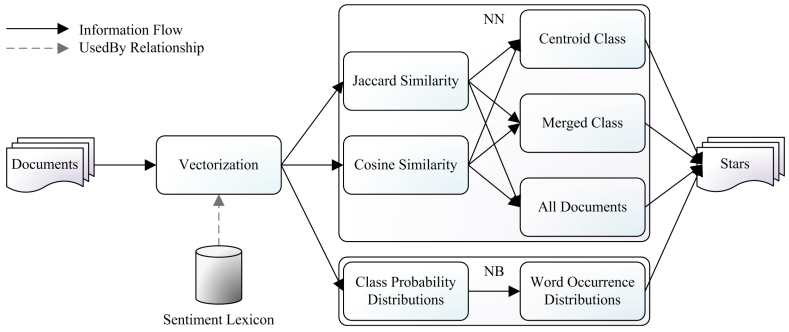

Fig. 1 Star rating classification of documents, represented by the occurrence of words retrieved from a sentiment lexicon, by means of nearest neighbor (NN) and naïve Bayes (NB) classifiers.

Our star rating classification approaches are summarized in Fig. 1. The nearest neighbor classifiers use the Jaccard or cosine similarity measure, combined with class representations based on the centroids, all documents, or a merger of all documents of a class. Our naïve Bayes classifier models document similarity by means of probability distributions for star rating classes, given documents, where each class is modeled as the probability distributions for word occurrences in that class.

4 Evaluation

The statistical sentiment analysis methods proposed in Sect. 3 can be used for classifying the star rating of reviews based on cues in the actual natural language content of these reviews. These cues are constituted by the occurrence of specific sentiment-carrying words, derived from a sentiment lexicon, and are reflected in our novel binary vector representations of reviews. Our proposed methods can be applied to these vectorized reviews in order to identify similarities between reviews and to exploit these, such that their associated star ratings can be determined.

4.1 Experimental Setup

In order to evaluate and compare our proposed star rating classification methods, we assess their performance on a data set containing reviews posted on Amazon [7]. In this data set, the reviews have been annotated by their respective authors with a star rating between one and five stars. The reviews cover a multitude of products, including books, music, and movies, and hence span multiple domains. We randomly sample 10,000 reviews from this data set as our training set, and 10,000 reviews as our test set. The reviews in both sets are approximately normally distributed over five star classes, while being somewhat skewed towards the higher ratings.

The reviews in our data set need to be represented by means of vectors signaling the presence of sentiment-carrying words. In our current endeavors, we extract these sentiment-carrying words from the Multi-Perspective Question Answering (MPQA) corpus [18], which contains a large collection of subjective words collected from several news sources, covering a wide variety of subjects. We extract all subjective words and subsequently discard all duplicate entries while not accounting for POS or meaning. This process leaves us with 4,300 lexical representations of sentiment-carrying words, i.e., 4,300 features for our binary vector representation of reviews.

We evaluate and compare the performance of several star rating classification approaches on our vectorized data. In our experiments, we consider the nearest neighbor and naïve Bayes approaches proposed in Sect. 3. For the nearest neighbor classifier, we consider both the Jaccard and the cosine similarity measure. Additionally, we consider class representations based on the centroid reviews of each class, all reviews, and a merger, i.e., union, of all reviews constituting each respective class.

Each method is assessed by means of several performance measures. First, we assess the average precision, recall, and F_1 measure over all five classes. Precision is the proportion of the reviews classified as, e.g., one star, which in fact should have been classified as such. Recall is the proportion of the reviews with a particular classification which are also classified as such. The F_1 measure is the harmonic mean of precision and recall. We also assess the overall accuracy, i.e., the percentage of correct classifications. Finally, we assess the Root Mean Squared Error (RMSE) of the class numbers in order to evaluate how far off the classifications typically are.

4.2 Experimental Results

The experimental results presented in Table 1 suggest that, on our data set and with our vectorization method of the reviews in this data set, the Jaccard similarity measure typically yields better results for the nearest neighbor method than the cosine similarity measure does, especially in terms of precision. Furthermore, in terms of precision, recall, and F_1 measure, a class representation based on all reviews in a particular class appears to outperform the other considered class representations. However, a merger of all vector representations of reviews constituting a particular class appears to yield better results in terms of overall accuracy and RMSE of assigned class numbers than a class representation based on all reviews does.

However, the nearest neighbor classifiers are clearly outperformed by the naïve Bayes star rating classifier, especially in terms of overall accuracy and RMSE of assigned class numbers. The naïve Bayes approach is however outperformed in terms of recall by nearest neighbor classifiers with a class representation based on all reviews in a particular class, yet this is compensated for by the relatively high precision and, to a lesser extent, F_1 measure of the naïve Bayes approach as compared to the nearest neighbor star rating classifiers. All in all, the naïve Bayes approach appears to be superior to all considered nearest neighbor approaches.

Table 1 Average precision, recall, F_1 measure, overall accuracy, and RMSE of assigned class numbers over all five star rating classes for the considered nearest neighbor (NN) and naïve Bayes (NB) star rating classifiers. The best performance is printed in bold for each performance measure.

Method	Precision	Recall	F_1	Accuracy	RMSE
NN (Jaccard, centroid)	0.241	0.235	0.219	0.300	1.879
NN (Jaccard, all)	0.294	**0.325**	0.261	0.323	1.673
NN (Jaccard, merged)	0.228	0.211	0.184	0.477	1.432
NN (cosine, centroid)	0.232	0.230	0.230	0.365	1.508
NN (cosine, all)	0.293	0.318	0.244	0.291	1.727
NN (cosine, merged)	0.227	0.229	0.223	0.392	1.567
NB	**0.328**	0.269	**0.269**	**0.508**	**1.296**

Even though the performance of some of our considered methods seems rather promising, the algorithms leave room for improvement. An error analysis has revealed that our considered approaches typically fail to correctly interpret more complex sentences, for instance those containing negation. Other errors are caused by occasionally sparse vectors due to a lack of identified sentiment-carrying words in some of the reviews in our data set. Another common source of errors appears to be off-topic noise in the reviews. People tend to discuss different aspects of their subjects and possibly even of other subjects before they arrive at their conclusions. The sentiment conveyed by the conclusions in such reviews appears to be a better proxy for the intended sentiment and thus for the overall verdict, quantified in a star rating. As such, a weighting scheme taking into account the position or role of words in a text may help improve the performance of our considered star rating classification methods.

5 Conclusions and Future Work

In this paper, we have proposed and assessed several statistical methods for classifying the star rating of reviews. The contribution of this work is two-fold. First, in an attempt to combine the classification accuracy and processing speed benefits of statistics-based sentiment analysis approaches with the robustness of lexicon-based approaches, we have proposed to represent the content of reviews by means of a binary vector representation, where the features represent the presence of sentiment-carrying words, retrieved from a general purpose sentiment lexicon. Second, we have compared the performance of several classifiers on these vector representations. A nearest neighbor classifier turns out to maximize recall, whereas a naïve Bayes classifier appears to excel in terms of precision, accuracy, and the RMSE of the assigned number of stars. These findings can help businesses in their marketing or reputation management efforts by providing a comparably reliable indication of intended sentiment in reviews. Such insights enable businesses to identify, e.g., why people like or dislike (aspects of) products or brands.

In future research, we plan to take more approaches into account in our comparisons of methods for star rating classification of sentiment. Furthermore, we plan to include additional features in our vector representations of content of reviews. Such features may be the frequencies, POS, and word senses of (sentiment-carrying) words. Additionally, we consider devising a weighting scheme for our vector representations in order to take into account the position or role of (sentiment-carrying) words in a text. Last, other machine learning algorithms, e.g., support vector machines, may be applied in order to possibly improve upon the performance of the star rating classification methods considered in our current endeavors.

Acknowledgements The authors of this paper are partially supported by the Dutch national program COMMIT.

References

1. Baccianella, S., Esuli, A., Sebastiani, F.: SentiWordNet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining. In: 7th Conference on International Language Resources and Evaluation (LREC 2010), pp. 2200–2204. European Language Resources Association (2010)
2. Heerschop, B., Goossen, F., Hogenboom, A., Frasincar, F., Kaymak, U., de Jong, F.: Polarity Analysis of Texts using Discourse Structure. In: 20th ACM Conference on Information and Knowledge Management (CIKM 2011), pp. 1061–1070. Association for Computing Machinery (2011)
3. Heerschop, B., van Iterson, P., Hogenboom, A., Frasincar, F., Kaymak, U.: Analyzing Sentiment in a Large Set of Web Data while Accounting for Negation. In: 7th Atlantic Web Intelligence Conference (AWIC 2011), pp. 195–205. Springer (2011)
4. Hogenboom, A., Hogenboom, F., Kaymak, U., Wouters, P., de Jong, F.: Mining Economic Sentiment using Argumentation Structures. In: Advances in Conceptual Modeling - Applications and Challenges, *Lecture Notes in Computer Science*, vol. 6413, pp. 200–209. Springer (2010)
5. Hogenboom, A., van Iterson, P., Heerschop, B., Frasincar, F., Kaymak, U.: Determining Negation Scope and Strength in Sentiment Analysis. In: 2011 IEEE International Conference on Systems, Man, and Cybernetics (SMC 2011), pp. 2589–2594. IEEE (2011)
6. Jansen, B., Zhang, M., Sobel, K., Chowdury, A.: Twitter Power: Tweets as Electronic Word of Mouth. Journal of the American Society for Information Science and Technology **60**(11), 2169–2188 (2009)
7. Jindal, N., Liu, B.: Opinion Spam and Analysis. In: 1st ACM International Conference on Web Search and Data Mining (WSDM 2008), pp. 219–230. Association for Computing Machinery (2008)
8. Melville, P., Sindhwani, V., Lawrence, R.: Social Media Analytics: Channeling the Power of the Blogosphere for Marketing Insight. In: 1st Workshop on Information in Networks (WIN 2009) (2009)
9. Paltoglou, G., Thelwall, M.: A study of Information Retrieval weighting schemes for sentiment analysis. In: 48th Annual Meeting of the Association for Computational Linguistics (ACL 2010), pp. 1386–1395. Association for Computational Linguistics (2010)
10. Pang, B., Lee, L.: A Sentimental Education: Sentiment Analysis using Subjectivity Summarization based on Minimum Cuts. In: 42nd Annual Meeting of the Association for Computational Linguistics (ACL 2004), pp. 271–280. Association for Computational Linguistics (2004)

11. Pang, B., Lee, L.: Opinion Mining and Sentiment Analysis. Foundations and Trends in Information Retrieval **2**(1), 1–135 (2008)
12. Pang, B., Lee, L., Vaithyanathan, S.: Thumbs up? Sentiment Classification using Machine Learning Techniques. In: Empirical Methods in Natural Language Processing (EMNLP 2002), pp. 79–86. Association for Computational Linguistics (2002)
13. Taboada, M., Brooke, J., Tofiloski, M., Voll, K., Stede, M.: Lexicon-Based Methods for Sentiment Analysis. Computational Linguistics **37**(2), 267–307 (2011)
14. Taboada, M., Voll, K., Brooke, J.: Extracting Sentiment as a Function of Discourse Structure and Topicality. Tech. Rep. 20, Simon Fraser University (2008). Available online, `http://www.cs.sfu.ca/research/publications/techreports/#2008`
15. van der Meer, J., Boon, F., Hogenboom, F., Frasincar, F., Kaymak, U.: A Framework for Automatic Annotation of Web Pages Using the Google Rich Snippets Vocabulary. In: Twenty-Sixth Symposium On Applied Computing (SAC 2011), Web Technologies Track, pp. 765–772. Association for Computing Machinery (2011)
16. Whitelaw, C., Garg, N., Argamon, S.: Using Appraisal Groups for Sentiment Analysis. In: 14th ACM International Conference on Information and Knowledge Management (CIKM 2005), pp. 625–631. Association for Computing Machinery (2005)
17. Wiebe, J., Wilson, T., Bruce, R., Bell, M., Martin, M.: Learning Subjective Language. Computational Linguistics **30**(3), 277–308 (2004)
18. Wiebe, J., Wilson, T., Cardie, C.: Annotating Expressions of Opinions and Emotions in Language. Language Resources and Evaluation **39**(2), 165–210 (2005)