A Context-Dependent Supervised Learning Approach to Sentiment Detection in Large Textual Databases

Albert Weichselbraun[1], Stefan Gindl[2] and Arno Scharl[2]

[1] Department of Information Systems and Operations, Vienna University of Economics and Business, Austria,
[2] Department of New Media Technology, MODUL University Vienna, Austria,
albert.weichselbraun@wu.ac.at, {stefan.gindl, arno.scharl}@modul.ac.at

Abstract. Sentiment detection automatically identifies emotions in textual data. The increasing amount of emotive documents available in corporate databases and on the World Wide Web calls for automated methods to process this important source of knowledge. Sentiment detection draws attention from researchers and practitioners alike - to enrich business intelligence applications, for example, or to measure the impact of customer reviews on purchasing decisions. Most sentiment detection approaches do not consider language ambiguity, despite the fact that one and the same sentiment term might differ in polarity depending on the context, in which a statement is made. To address this shortcoming, this paper introduces a novel method that uses Naïve Bayes to identify ambiguous terms. A contextualized sentiment lexicon stores the polarity of these terms, together with a set of co-occurring context terms. A formal evaluation of the assigned polarities confirms that considering the usage context of ambiguous terms improves the accuracy of high-throughput sentiment detection methods. Such methods are a prerequisite for using sentiment as a metadata element in storage and distributed file-level intelligence applications, as well as in enterprise portals that provide a semantic repository of an organization's information assets.

Categories and Subject Descriptors: H. Information Systems [**H.3 Information Storage and Retrieval**]: H3.1 Content Analysis and Indexing, H3.3 Information Search and Retrieval

Keywords: annotation, document enrichment, machine learning, natural language processing

1. INTRODUCTION

Sentiment detection, which is often also referred to as *sentiment analysis*, *sentiment classification*, or *opinion analysis/mining* extracts sentiment by rating a segment of text as either positive (favorable) or negative (unfavorable). This allows investigation into how the author of a document perceives a certain product, service, tourism location, or political party.

Sentiment in the form of semantic annotations is highly relevant for corporate database applications such as enterprise portals, media archives, and information lifecycle solutions. The storage industry has been traditionally focused on reducing complexity around applications and hardware, and only recently recognized the need for intelligent processing of unstructured data. A scalable architecture for annotating unstructured data therefore fills an important gap in a rapidly expanding market.

The applicability of sentiment data is not restricted to corporate database applications. The World Wide Web offers a vast number of communication platforms such as forums, blogs and product review sites which act as a focal point for interactive opinion exchange. In addition to these visible sources, nearly half a million databases are hidden behind query forms in the largely unexplored frontier of the Deep Web [He et al. 2007]. Textual data from these sources, as summarized in Figure 1, is readily available and can serve as a valuable repository of consumer opinion (e.g., as benchmark for a company's products and services). Negative postings can be an indicator of poor quality or a bias towards other products and services, which should trigger product improvements or changes to the marketing strategy. Politicians and decision makers can also benefit from sentiment analysis as it provides means to gather indirect feedback on the public's perception [Scharl and Weichselbraun 2008] and complements direct methods, such as opinion polls.

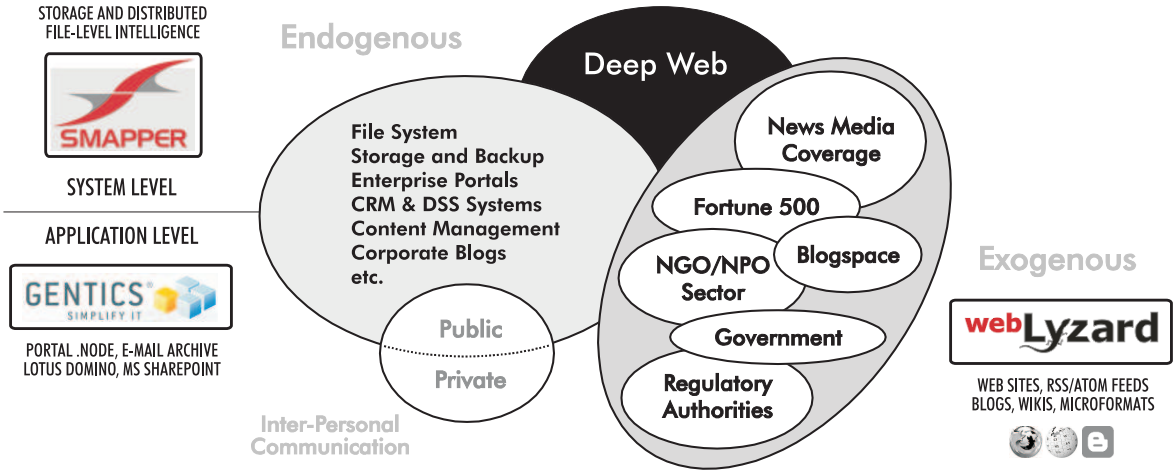

Fig. 1. Sources of textual data (corporate database applications, third-party Web sites, Deep Web queries)

Pang and Lee [2008] demonstrate the importance of customer reviews and show that between 73 and 87% of the readers of online reviews of restaurants, hotels and travel services reported that these reviews had a significant impact on their purchase decision. Consumers also report that they are willing to pay considerably more (between 20 and 99% depending on the product) for higher-rated products [Pang and Lee 2008]. The growing importance of sentiment analysis is also reflected in the growing attention this research area has received in recent years.

Processing textual data to detect sentiment automatically remains a challenging task, even when utilizing part-of-speech (POS) tagging and other common text processing methods. Ambiguity and subtle incremental change of tonal expressions between different versions of a document complicate the detection of its sentiment and often prevent promising algorithms from revealing their full potential. This paper addresses the issue of context-dependent ambiguities by introducing a novel approach to create and use contextualized dictionaries for sentiment detection. The presented approach uses a sentiment lexicon as a basis for sentiment detection. It refines the sentiment values of the lexicon terms depending on their usage context. The Naïve Bayes technique builds the mathematical background for the context-dependent calculation of the sentiment values. Naïve Bayes, as a simple but powerful technique, perfectly fits to prove the hypothesis that contextualization helps improving lexicon-based sentiment detection.

The remainder of this paper is structured as follows: Section 2 summarizes state-of-the-art techniques in sentiment detection. We then outline our approach and how it can be applied to detect sentiment (Section 3). Section 4 presents an extensive evaluation of the context-aware sentiment detection component. The paper closes with an outlook and conclusions in Section 5.

2. RELATED WORK

Early work on sentiment detection started with the identification of subjective sentences [Wiebe 1994] and the discrimination of positive and negative adjectives by exploiting the mutual information of known sentiment indicators and unknown adjectives by analyzing their syntactical relations [Hatzivassiloglou and McKeown 1997].

The used techniques can be roughly divided into three areas: lexical approaches, machine-learning approaches, and combinations of the two former. Lexical approaches use so-called "sentiment lexicons" (opinion lexicons or tagged dictionaries). These are lists of known sentiment terms, where each term

has a *sentiment value* (mostly a numerical value ranging from $[-1, 1]$) assigned to it. For example, the term *excellent* is an intuitively positive sentiment term which would have the value 1. A term such as *terror*, commonly indicating something negative, would have the value -1 assigned to it. Pure machine learning approaches do not rely on those lexicons but invoke other features to accomplish sentiment detection. Mixed approaches combine lexical with machine learning techniques. The presented work is such a mixed approach. In the following sections we provide an overview of sentiment detection approaches relevant to our work and illuminate their application areas. For additional information and sources please also see the detailed survey of sentiment detection written by Liu [2010].

2.1 Sentiment Detection

Many approaches rely on the classification of entire documents to evaluate their techniques. A very popular application area are customer reviews such as movie [1], product [2], or destination reviews [3]. These reviews have the advantage of already being classified by the authors who assign a rating, subsuming the overall sentiment of the review in a single score. For instance, Amazon allows customers to review all of their products using a five star scale and TripAdvisor facilitates destination scoring for tourists based on five circles. Reviews with less than three stars (circles, respectively) can be considered as negative judgments, more than three stars (circles) are positive judgments; a number of three indicates neutral attitude. Moreover, customer reviews can be easily accessed by employing web crawlers to automatically capture large numbers of them.

Many sentiment detection approaches provide a binary classification, i.e. they determine if a review expresses positive (more than three stars) or negative (less than three stars) sentiment. Pang et al. [2002] applied different machine learning approaches such as Naïve Bayes, Support Vector Machines and Maximum Entropy Modeling on movie reviews and obtained considerable results. These machine learning techniques are well-known from topic categorization, yet they could not deliver as satisfactory results for sentiment detection as they can for the categorization of topic. The authors believe that there are more subtle features necessary to unfold the full potential of these methods [Pang et al. 2002].

Turney [2002] performs binary classification on product reviews. Like Hatzivassiloglou and McKeown [1997] he uses a lexicon containing a set of known sentiment terms which he extends by applying Pointwise Mutual Information (PMI) and Latent Semantic Analysis (LSA). The work shows that a simple technique such as PMI is able to outperform the more complicated LSA in such settings. A more fine-grained approach presented by Pang and Lee [2005] determines the exact number of stars provided by the review author. Three methods (one versus all, regression, and metric-labeling) based on Support Vector Machines accomplish this task. To assess the feasibility of such a fine-grained analysis, the authors conducted a manual evaluation to demonstrate that humans are indeed capable of determining an exact star rating and are not limited to binary decisions.

Beineke et al. [2004] refine Turney's work [Turney 2002] by applying a Naïve Bayes model which they train on a labeled and an unlabeled corpus. Like Turney, they use a list of seed terms for the classification of new words, which only contains five positive and negative sentiment terms, as well as a larger list which they assemble from the WordNet [Fellbaum 1998] synonyms of the terms *good, best, bad, boring*, and *dreadful*. The authors conclude that their method outperforms previous approaches in regard to classification accuracy and speed of computation.

Dave et al. [2003] compare the efficiency of a simple term-counting algorithm with different machine-learning algorithms. For that purpose they crawled a large number of electronics product reviews from CNET and reviews of books, movies, music and products from Amazon. They divide these reviews

[1] www.imdb.com
[2] www.amazon.com
[3] www.tripadvisor.com

into a test and a training set and use a simple frequency based approach as a baseline which they compare with machine-learning implementations, such as Naïve Bayes and Support Vector Machines. They also try several Information Retrieval techniques to pre-process the data. The simple baseline delivers results similar to results obtained by machine learning algorithms, which demonstrates that simple techniques should not be underestimated.

Subrahmanian and Reforgiato [2008] examine the impact of adverb-verb-adjective combinations. They define a number of axioms describing how they influence each other. In their experiments, Subrahmanian and Reforgiato achieve promising results on 200 news pages. Nicholls and Song [2009] examine the impact of different part-of-speech tags by employing a Maximum Entropy classifier. They consider only adverbs, adjectives, verbs and nouns as relevant for sentiment detection and assign these categories different weights. According to their results, adjectives and adverbs are the strongest sentiment conveyors, while verbs and nouns contribute only little.

2.2 Considering Context Information

Context information is an essential ingredient for assessing the meaning of textual information. This section summarizes various approaches to utilizing context information for improving sentiment detection algorithms.

Nasukawa and Yi [2003] describe sentiment detection as a three step process: (1) the identification of sentiment expressions, (2) the determination of polarity and strength of the expressions and (3) the relationship of the sentiment expressions to their subject. Verbs indicate relationships and can either directly affect an argument (i.e. a target term) or transfer sentiment from one argument to the other. With such a model the authors are able to handle expressions like t_i *prevents trouble*. In that example, the verb *prevents* transfers the opposite sentiment of argument *trouble* to the term t_i. Terms with part-of-speech tags different from 'verb' are treated in a simpler way - they directly transfer their sentiment to the related argument.

Wilson et al. [2005] consider context by applying a filtering process which uses context information based on 28 features. Afterwards, they determine the polarity of the remaining sentences by considering a total of ten features. These features of both steps are trained and tested with BoosTexter's AdaBoost.MH algorithm [Schapire and Singer 2000], which identifies sentiment expression in the Multi-perspective Question Answering (MPQA) Opinion Corpus[4] [Wiebe et al. 2005]. The evaluation shows that both polar-neutral filtering and polarity classification benefit from using the proposed features. Wilson et al. [2009] expand this approach by using four different machine learning algorithms – BoosTexter's Adaboost.HM, the rule-based learner Ripper [Cohen 1996], TiMBL [Daelemans et al. 2001] for memory-based learning and an SVM implementation [Joachims 1999]. The authors evaluate their system using an extended part of the MPQA corpus. The findings of their work show that neutral-polar filtering is important and that large feature sets are necessary to accomplish both neutral-polar filtering as well as polarity classification.

Polanyi and Zaenen [2006] address several issues on context recognition and propose handling strategies from a linguistic point of view. They divide concepts responsible for context switches into two groups: *Sentence Based Contextual Valence Shifters* and *Discourse Based Contextual Valence Shifters*.

SentiWordNet [Esuli and Sebastiani 2006], a sentiment resource based on WordNet, also uses context invocation by propagating sentiment values across synset terms. They use a semi-supervised approach to classify all WordNet synsets into positive, negative and objective. At first, they manually label all synsets containing 14 paradigmatic terms, creating 47 positive and 58 negative synsets. All synsets having a connection to these seed synsets are labeled accordingly. Used relations are *direct antonymy*, *similarity*, *derived-from*, *pertains-to*, *attribute*, and *also-see*. Afterwards, they identify ob-

[4]nrrc.mitre.org/NRRC/publications.htm

jective synsets as those synsets which are not in the previously identified bag of synsets and which contain objective terms according to the General Inquirer. These three sets serve as training data to train eight ternary classifiers, which then classify the remaining parts of WordNet.

Our approach is fundamentally different from the presented techniques. We do not handle a sentiment transfer from sentiment terms to subjects [Nasukawa and Yi 2003], nor do we have a polar/neutral filtering or predefined syntactical features [Wilson et al. 2005; 2009]. Instead, the proposed method considers the term's context based on discriminators identified in the text and adjusts its sentiment values accordingly. We also use explicit linguistic features to detect negations but do not use sentiment inheritance with linguistic relations such as synonymy [Esuli and Sebastiani 2006].

3. METHOD

A so-called "sentiment lexicon" serves as the basis for our approach. The sentiment lexicon is a collection of sentimental words, e.g. *excellent* or *bad*. We use a sentiment lexicon derived from the General Inquirer's sentiment list [Stone et al. 1966]. To achieve higher term coverage we applied reverse lemmatization on the original terms, i.e. for each term in the General Inquirer we also added its inflected forms. For example, for the term *celebrate* we added *celebrated*, *celebrates*, and *celebrating*. The method introduced in this work addresses the problem that certain sentiment terms change their sentiment value depending on the context. The example below demonstrates this problem based on the sentiment of the term *repair*:

— "After the *repair* by Leica the camera operates superbly."
— "You are saved the bother of shipping a defective unit back to the *repair* station."
— "Authorized Nikon *repair* shops charge more than the cost of a new camera."

The term - sentiment distributions in Figure 2 illustrate how a term's sentiment may change with its context. The diagrams on the left side represent ideal term distributions, whereas the right side contains distributions of real terms as found in our database. The left upper graph demonstrates a term with an unambiguous negative sentiment - such a term will only occur in strongly negative reviews. In contrast, a neutral term (second graph on the left side) is more or less evenly distributed regardless of the text's sentiment. Ambiguous terms such as the one illustrated in the third and fourth graphic show multiple maxima. Their usage depends on external factors such as the term's context (see the example above) or its part-of-speech tag (e.g. in the sentences 'People like you and me' and 'Peter likes swimming' the term *like* is once used as a preposition and once as a verb, resulting in different meanings of one and the same term). The graphs on the right side of Figure 2 show examples of real terms. *Worst*, as an unambiguous negative term, has one clear peak at the left side. *And*, as a neutral term, has nearly equal frequencies on both polarity sides. *Accident* is an ambiguous and polarizing term, indicated by two peaks at the leftmost and rightmost side of its graph. Interestingly, *expensive* turns out to be an ambiguous term as well, although intuition would suggest a negative association. This fact shows that even very "safe" sentiment terms can occur in unexpected contexts.

Discriminating contexts where a sentiment term is positive from those where the same sentiment term is negative improves the accuracy of sentiment detection. The approach presented in this paper determines context-dependent sentiment values for ambiguous sentiment terms by:

(1) identifying sentiment terms (t_i), whose sentiment changes with the context,
(2) using the Naïve Bayes algorithm to determine potential discriminators (c_i) which help distinguish between the term's usage in a positive (C^+) or negative (C^-) context.
(3) learning the probabilities $p(C^+|c_i)$ and $p(C^-|c_i)$ that a term's discriminators (c_i) suggests a positive (C^+) or negative (C^-) context.
(4) considering the term's context for terms which sentiment value does vary significantly with its usage context.

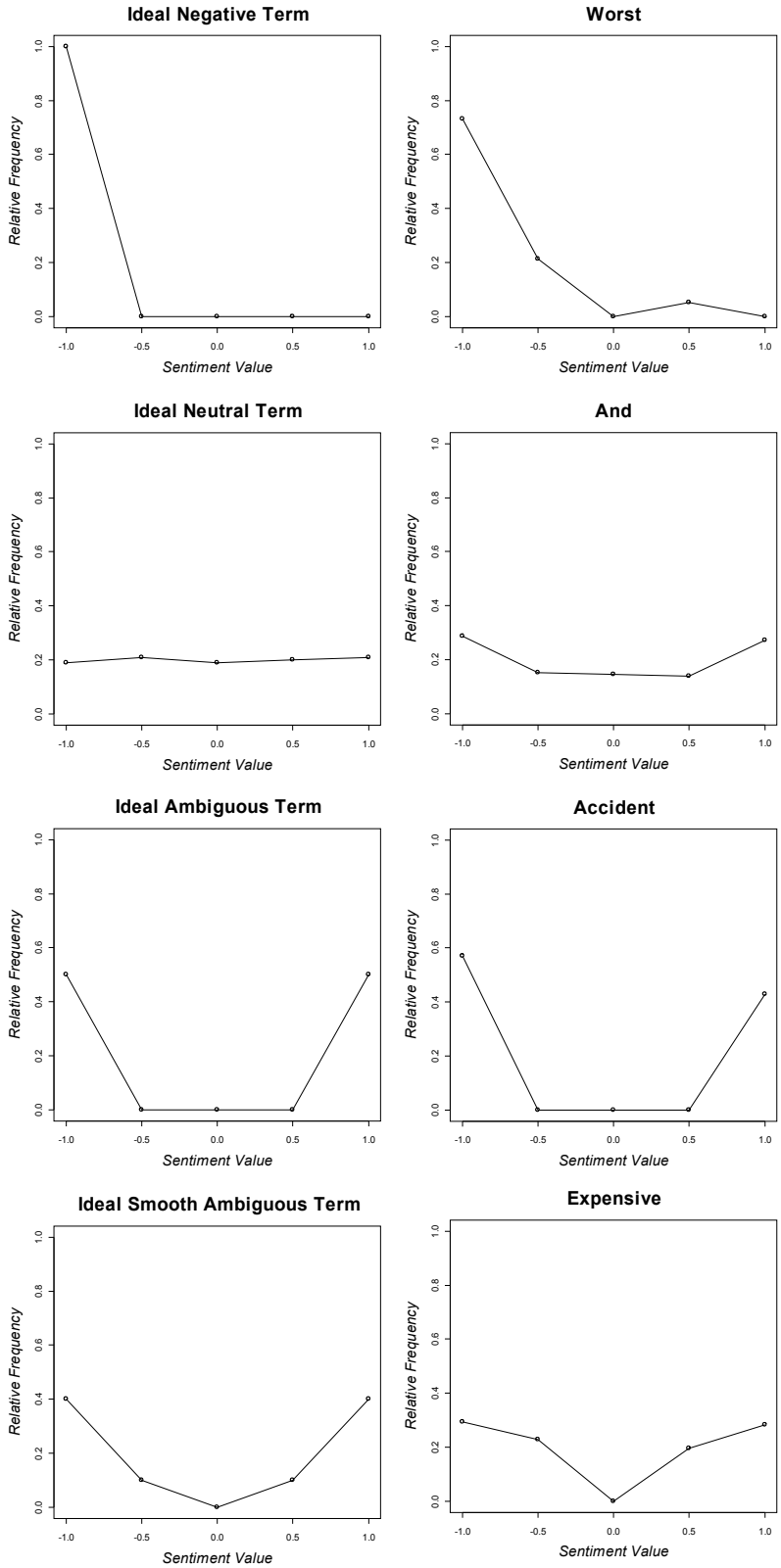

Fig. 2. Comparison of the frequency graphs of ideal and real terms

Figure 3 outlines the system architecture of the proposed approach. The sentiment detection process is divided into three individual, consecutive parts (the first two steps create the knowledge base, i.e. the contextualized sentiment lexicon, providing the information necessary to determine the sentiment in the last step):

(1) **Ambiguous Term Determination:** The system identifies ambiguous sentiment terms based on a training corpus.
(2) **Context Term Determination:** Given the previously identified ambiguous terms, the system collects terms co-occurring with each ambiguous term (i.e. context terms) and computes the probabilities required for the Naïve Bayes algorithm. Currently, the system takes all terms occurring in a document as context terms. Future work will support smaller windows sizes such as paragraphs or sentences in this contextualization step (Section 5). A new lexicon, the so-called "*contextualized sentiment lexicon*", stores all ambiguous terms as well as their context terms. For each context term, it also stores the frequency of it in positive and negative documents.
(3) **Sentiment Detection:** Given some new, unclassified document the system computes the overall sentiment of the document based on the unambiguous terms in the sentiment lexicon and the ambiguous terms in the contextualized sentiment lexicon by considering their context.

3.1 Preprocessing of the Corpus

Both training and test corpus (their properties are explained in more detail in Section 4) are preprocessed in the same way. All text in the reviews is tokenized by a simple word boundary tokenizer. Additionally, we detect negated terms according to a list of 23 negation triggers. A term is negated when it directly follows a trigger.

3.2 Identifying Ambiguous Terms

Based on a sentiment term's distribution in positive and negative reviews we compute it's average sentiment value (μ_i) and the standard deviation of its sentiment value (σ_i). Extensive experiments have shown that the method performs well, if terms t_i with

$$\sigma_i \geq 0.75 \quad \text{and} \tag{1}$$
$$\mu_i + \sigma_i \geq 0.25 \quad \text{and} \tag{2}$$
$$\mu_i - \sigma_i \leq -0.25 \tag{3}$$

are considered ambiguous. Ambiguous terms are therefore sentiment terms which show a high enough standard deviation (Equation 1) which is able to shift the term's meaning (Equation 2-3). This guarantees the term being represented in both polarity classes with sufficiently high frequency.

3.3 Retrieving Context Terms

We locate all terms (c_i) co-occurring with the ambiguous terms identified above and determine their probabilities of occurring in positive $p(C^+|c_i)$ and negative $p(C^-|c_i)$ reviews as well as in the whole corpus $p(c_i)$. The system saves the information on the terms' discriminators and their probabilities in the *contextualized sentiment lexicon*.

3.4 Sentiment Detection

The sentiment detection method uses the sentiment lexicon to determine the text's sentiment. The sentiment contributed by ambiguous terms is computed by determining the term's context **c** using

the Naïve Bayes algorithm on all context terms c_i available in the review.

$$\mathbf{c} = \{c_1, ...c_n\} \tag{4}$$

$$p(C^+|\mathbf{c}) = \frac{p(C^+) \cdot \prod_{i=1}^{n} p(c_i|C^+)}{\prod_{i=1}^{n} p(c_i)} \tag{5}$$

Following a suggestion by Zdziarski [2005] we only consider the ten most significant discriminators ($n = 10$) in the equation above, i.e. those context terms having the largest deviation from a neutral (0.5) probability value. In our calculation, positive sentiment is expressed by the value 1 and negative sentiment by -1. The overall sentiment of a document is the sum of the sentiment of all terms (t_i) occurring in that particular document considering both unambiguous and ambiguous sentiment terms.

$$s_{total} = \sum_{t_i \in doc} n(t_{i-1})[s(t_i) + s'(t_i|C)] \qquad \text{with} \tag{6}$$

$$n(t_{i-1}) = \begin{cases} -1.0 & \text{if } t_{i-1}\text{indicates a negation} \\ +1.0 & \text{otherwise.} \end{cases} \tag{7}$$

The function $s(t_i)$ returns a term's sentiment value or zero if the sentiment lexicon does not contain the term. $s'(t_i)$ considers the contextualized sentiment lexicon and returns a term's contextualized sentiment score or zero if the term is not present in the dictionary. Sentiment terms either occur in the sentiment lexicon $s(t_i) \neq 0$ or in the contextualized sentiment lexicon $s'(t_i) \neq 0$. The function $n(t_{i-1})$ detects negations and adjusts the sentiment score accordingly.

4. EVALUATION

We accomplished the evaluation on three different evaluation sets based on 2 500 customer reviews from Amazon, 1 800 reviews from TripAdvisor, and the movie review corpus used in [Pang and Lee 2004], consisting of 2 000 reviews. All corpora consisted of an equal number of positive (>3 stars) and negative (<3 stars) reviews. We accomplished a 10-fold cross validation, without randomizing training and test sets, thus both sets are completely disjunct. This prevents a pollution of the test results by reviews occurring both in the training and test set. As baseline served a lexical algorithm. It counts the number of positive and negative sentiment terms (i.e. terms contained in the sentiment lexicon described in Subsection 3) in a document. If the number of positive terms outweighs the number of negative terms the document is assigned a positive overall sentiment value, and vice versa.

In the following, we first outline the detailed evaluation results, give examples on observed ambiguous terms with context terms, and finally compare our research to already existing work.

4.1 Results

On the TripAdvisor corpus we observed very promising results. Table I contains the average values of Recall, Precision and F-Measure on all evaluation runs for both baseline and Naïve Bayes contextualization. It also contains significance values obtained with the R[5] implementation of Wilcoxon's rank sum test which shows the significance of the improvement compared to the baseline. Figure 4 contains a graphical overview of all evaluation runs. A checkmark indicates significant improvement; a dot marks non-significance or losses. Confidence values below 5% (i.e. $p < 0.05$) can be regarded as significant.

We achieved five significant gains on the TripAdvisor corpus; only the gain for positive Recall remained not significant. Thus, we consider the results on the TripAdvisor corpus as a success of Naïve Bayes contextualization.

[5]www.r-project.org

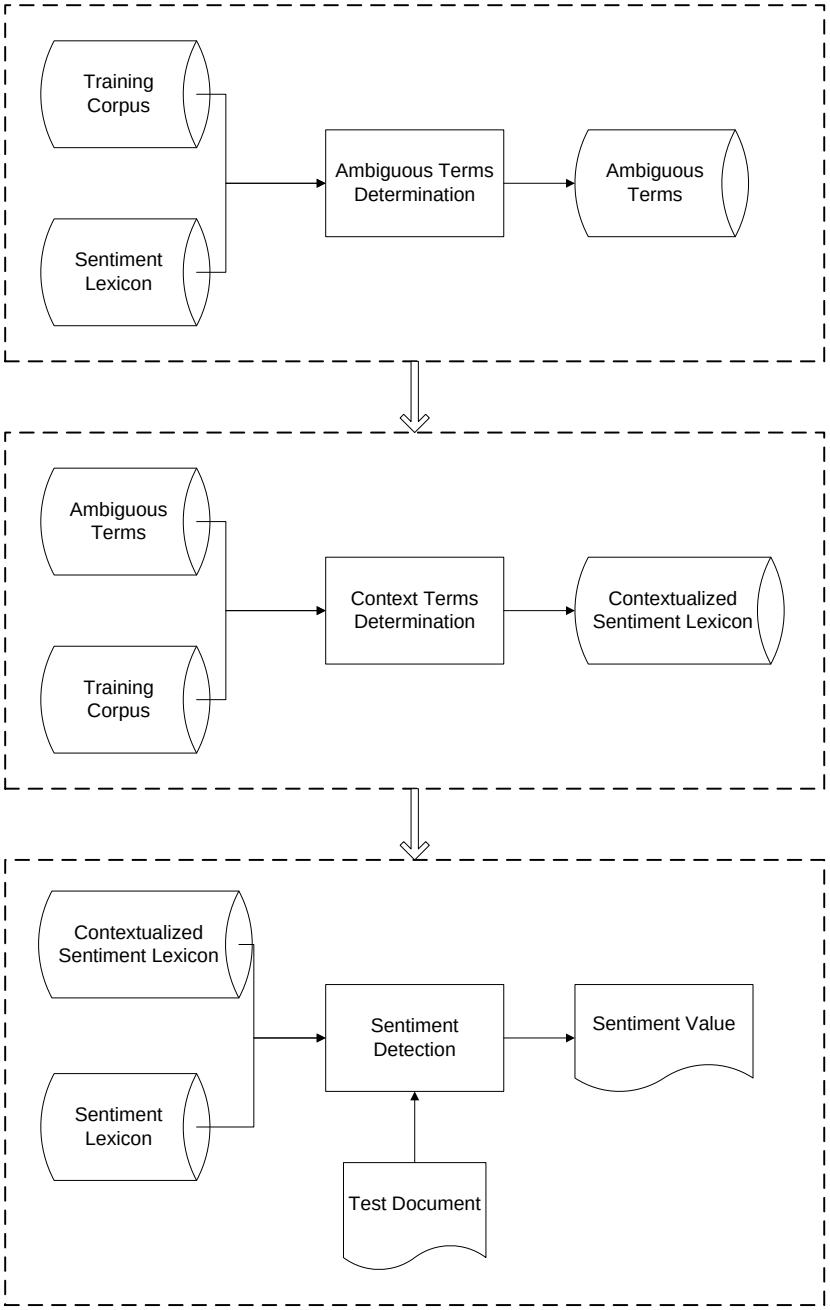

Fig. 3. Creation and application of a contextualized sentiment lexicon

	Baseline			NB			Significance		
	$\overline{R}$	$\overline{P}$	$\overline{F}_1$	$\overline{R}$	$\overline{P}$	$\overline{F}_1$	p_R	p_P	p_{F_1}
Pos	0.96	0.60	0.74	0.97	0.66	0.79	· (0.31)	✓(0.01)	✓(0.01)
Neg	0.34	0.90	0.49	0.46	0.95	0.61	✓(0.00)	✓(0.04)	✓(0.01)

Table I. The average results of the 10-fold cross validation on the TripAdvisor dataset

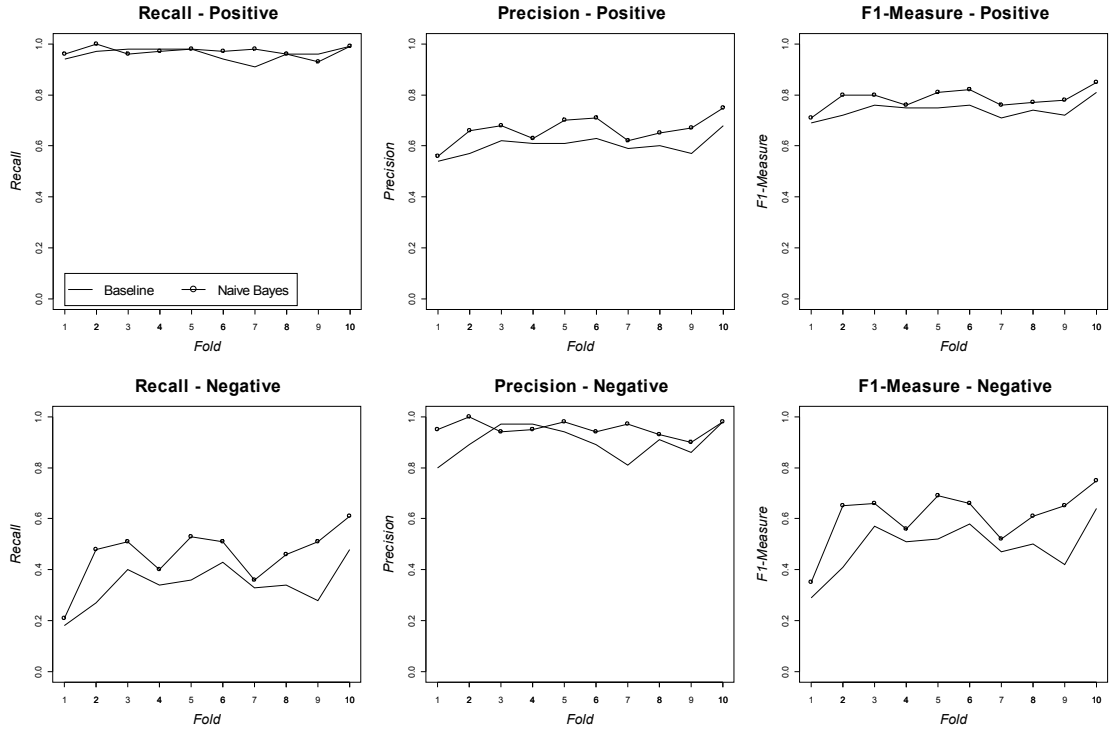

Fig. 4. Graphical overview over all cross-validation results (Test corpus: TripAdvisor).

On the Amazon corpus we could also prove the efficacy of Naïve Bayes contextualization. Table II shows the average results for both baseline and contextualization on all ten runs of the evaluation. As before, the table also contains significance values. The results show a significant improvement of precision in the detection of positive reviews, with a slight, statistically insignificant loss in recall. Figure 5 illustrates the detailed results of the cross-validation runs. For the detection of negative reviews the proposed approach shows its full strength. Both recall and F_1-Measure increase significantly; there is also an improvement in precision, although statistically insignificant.

	Baseline			NB			Significance		
	$\overline{R}$	$\overline{P}$	$\overline{F}_1$	$\overline{R}$	$\overline{P}$	$\overline{F}_1$	p_R	p_P	p_{F_1}
Pos	0.80	0.64	0.71	0.75	0.75	0.74	· (0.54)	✓(0.00)	· (0.13)
Neg	0.53	0.74	0.62	0.71	0.79	0.73	✓(0.00)	· (0.13)	✓(0.00)

Table II. The average results of the 10-fold cross validation on the Amazon dataset

The third corpus is the movie review corpus presented in [Pang and Lee 2004]. The results achieved with the contextualized lexicon are superior to those we had using the baseline. Especially for detecting negative reviews, the gains are remarkable. Yet, we recorded a decrease in recall for positive reviews. Table III shows the results of the 10-fold cross-validation on this dataset. Our approach achieved an accuracy of 76.6 %. Pang and colleagues achieved 86.4 % accuracy using Naïve Bayes; with Support Vector Machines they achieved 87.15 %. These results are considerably higher than the results of our approach. We ascribe this to two facts: (i) Pang et al. used a more sophisticated pre-processing than our system. They actually used two classifiers, one filtering objective sentences out of their dataset. The subjective sentences remaining from this objectivity filtering were then used for polarity classification. Our system will also benefit from such sophisticated pre-processing steps, which will be

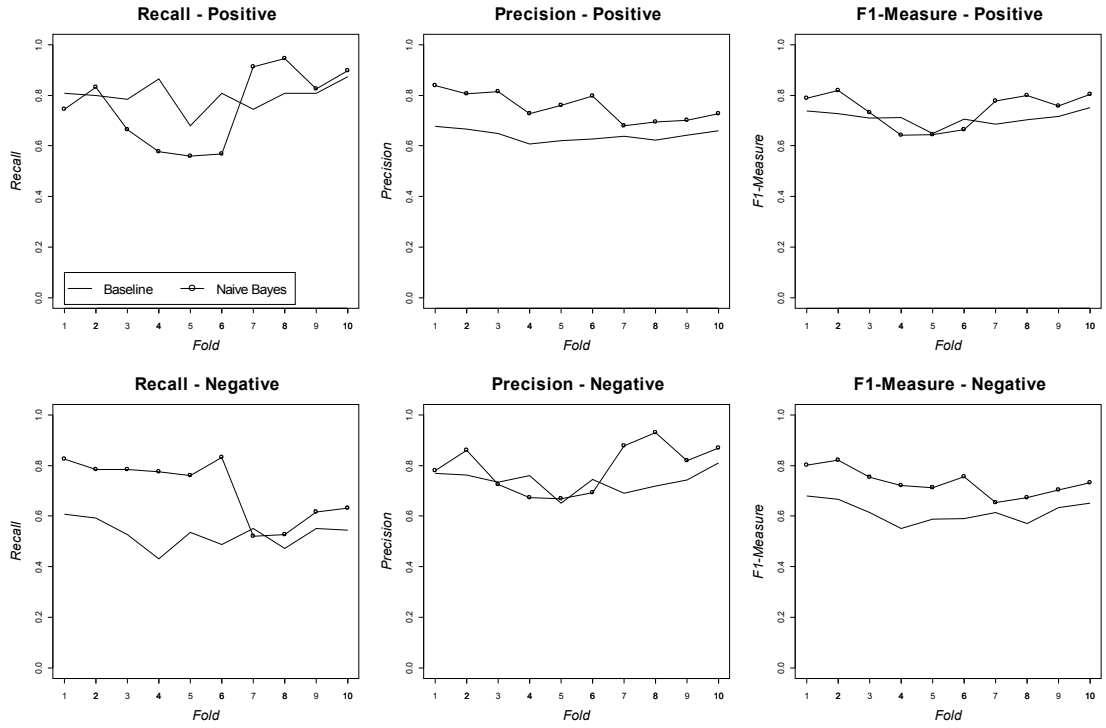

Fig. 5. Graphical overview over all cross-validation results (Test corpus: Amazon).

included in future work (see Section 5). (ii) The reviews in the movie corpus used by Pang et al. are huge compared to the reviews of our other corpora. Thus, the chosen context window size (currently the whole document) is too large and adds too much noise to the contextualized lexicon.

Furthermore, the contextualized lexicon is subject to the following trade-off: it is intended to overcome a problem of machine-learning techniques, which is their domain-dependency. Thus, the contextualized lexicon does not perform as well as a classifier specifically trained for that particular domain, but has the advantage of being applicable across domains. We plan to extract features for our contextualized lexicon in a way that they become applicable to documents of a domain different from the one used for training. A preliminary evaluation confirms the cross-domain applicability of such contextualized sentiment lexicons. We used the Naïve Bayes classifier of the Natural Language Toolkit[6], trained it on either (i) the Amazon or (ii) the TripAdvisor dataset and tested it on the other dataset. We did the same with the contextualized lexicon. With this strategy the Naïve Bayes classifier achieved accuracies of 59% (test set: Amazon) and 70% (test set: TripAdvisor); the contextualized lexicon had accuracies of 71% and 74%, respectively.

	Baseline			NB			Significance		
	$\overline{R}$	$\overline{P}$	$\overline{F}_1$	$\overline{R}$	$\overline{P}$	$\overline{F}_1$	p_R	p_P	p_{F_1}
Pos	0.69	0.63	0.66	0.59	0.91	0.72	· (0.01)	✓(0.01)	✓(0.01)
Neg	0.6	0.66	0.63	0.94	0.7	0.8	✓(0.01)	✓(0.01)	✓(0.01)

Table III. The average results of the 10-fold cross validation on the Movie dataset

[6]www.nltk.org

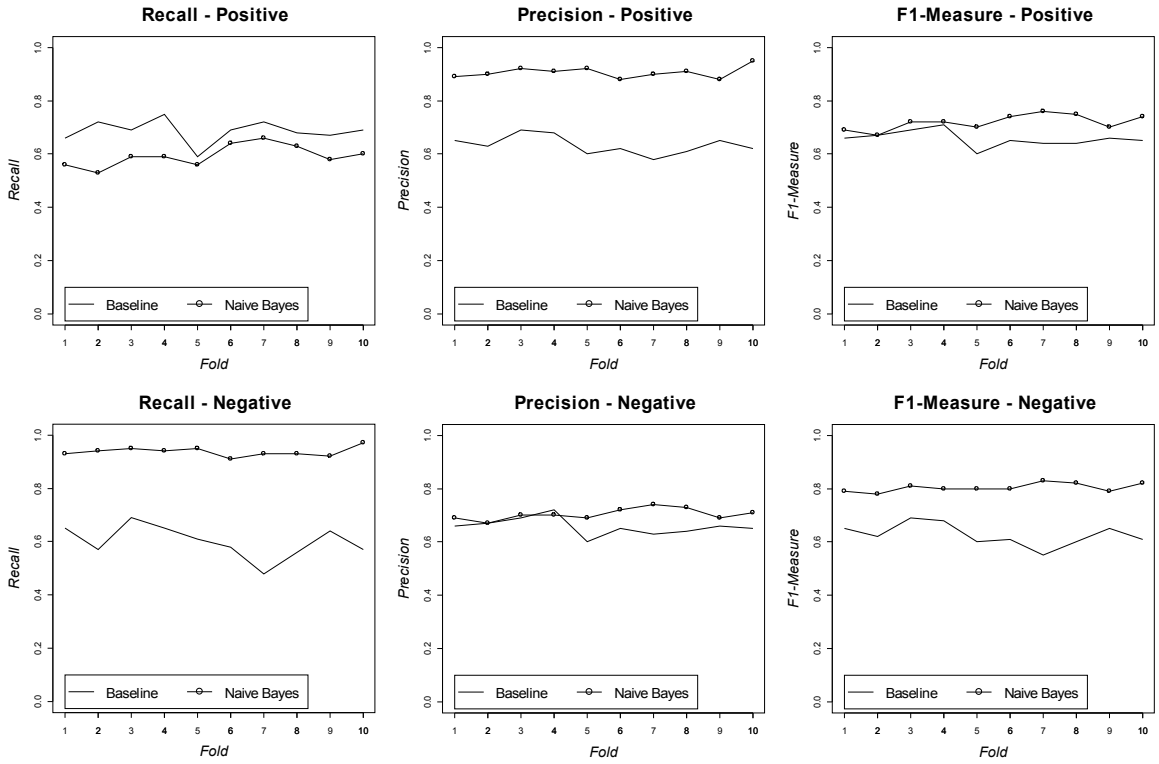

Fig. 6. Cross-validation results of the Movie corpus

In conclusion, the evaluation confirmed the efficacy of the proposed contextualization. It supports the hypothesis that context plays an important role in lexicon-based sentiment detection and that the Naïve Bayes technique can be used to properly assess a sentiment term's context. The simplicity of Naïve Bayes allows for an easy implementation, and its powerfulness provides reliability and convincing results. The improvements are stable, except for a non-significant loss in recall on the Amazon corpus and a decrease in recall on the Movie dataset. The gain in precision compensates this loss. The evaluation showed that the presented approach is powerful and promising, significantly outperforming the baseline on both datasets. In the next Subsection we provide some examples to demonstrate, how context can possibly change the polarity of sentiment terms.

4.2 Illustration

In the following table we provide some meaningful examples that we observed during evaluation. Table IV lists three sentiment terms whose polarity had been inverted by context terms, including a sentence for each term to illustrate the process. In the first line, the context term *complex* is an indicator for positive context, when the ambiguous term **burden** occurs. Thus, its originally negative value of -1 (according to the sentiment lexicon) is turned into +1. **Complaint**, originally also a negative term, turns into a positive term in combination with *small*. **Correct**, on the other hand, turns out to be a negative term when it has *problem* as a context term. The presented sentences are just review snippets. There are more context terms available in the rest of the review; nevertheless, these snippets give a good insight into the switches in sentiment values observed in practice.

t_{ambig}	$SV_{original}$	$t_{context}$	**Example Sentence**
burden	-1	complex	It'd be nice if it went to 85mm or so, but then it would probably have to be a little more complex, slower, and heavier, so why burden the design?
complaint	-1	small	The only very small complaint that I would have is that there is a bit of red eye, even with the reduction, amongst people who have blue eyes.
correct	1	problem	But don't buy one unless you can verify that something has been done to correct the handle problem.

Table IV. Examples of sentiment terms, where the context switches the term's polarity. $SV_{original}$ denotes the value of t_{ambig} according to the sentiment lexicon. This value is inverted because of the context of the sentence

5. CONCLUSION AND OUTLOOK

This paper presents a novel approach to sentiment detection in large textual databases that addresses the problem of polarity shifts of terms depending on their usage context. Identifying such shifts allows for a more focused processing of unstructured data from corporate database applications, third-party Web sites, and Deep Web queries. A formal evaluation demonstrates the importance of considering context and the advantages of using Naïve Bayes for creating and applying contextualized sentiment lexicons. When properly trained on pre-classified documents, such a procedure can significantly improve the quality of automated sentiment detection.

Several issues have to be addressed in future work: as mentioned in Section 3, the current window size for context is the whole document, which clearly needs refinement. Reviews are usually shorter than other documents (e.g. newspaper articles). The large size of the current window most likely diminishes the quality of our approach. It might be more useful to narrow the window to the paragraph, multi-sentence or even single-sentence level. Another issue is the refinement of pre-processing. A strategy for smart filtering of useless context terms (e.g. based on their occurrence frequency) will reduce the size of the contextualized lexicon and improve its quality. Another issue is the employment of different machine learning techniques. Currently we use the Naïve Bayes method because it is simple to implement but still delivers good results. Other techniques might deliver superior results. Support Vector Machines are especially known as a powerful tool for the classification of textual data.

The algorithm's training process was based on review corpora of fairly constrained domains. This tailors the procedure to a particular domain and reduces the applicability of the resulting lexicons in generic settings, or when attempting to process documents originating from a different domain. Future work will address this problem and distinguish between domain-specific and domain-independent context terms, improving the cross-domain applicability of this approach. We expect additional gains in performance by the invocation of phrases (n-grams) either as sentiment phrases with initial polarity or as context indicators.

ACKNOWLEDGMENT

This work was developed as a part of the RAVEN (Relation Analysis and Visualization for Evolving Networks; www.modul.ac.at/nmt/raven) research project funded by the Austrian Ministry of Transport, Innovation & Technology (BMVIT) and the Austrian Research Promotion Agency (FFG) within the strategic objective FIT-IT (www.fit-it.at).

REFERENCES

Beineke, P., Hastie, T., and Vaithyanathan, S. The Sentimental Factor: Improving Review Classification via Human-provided Information. In *Proceedings of the Annual Meeting on Association for Computational Linguistics*. Morristown, USA, pp. 263–269, 2004.

Cohen, W. W. Learning Trees and Rules with Set-valued Features. In *Proceedings of the National Conference on Artificial Intelligence*. Portland, USA, pp. 709–717, 1996.

Daelemans, W., Zavrel, J., and van der Sloot, K. TiMBL: Tilburg Memory Based Learner version 5.0 Reference Guide. Tech. rep., ILK Technical Report 03-10, Induction of Linguistic Knowledge Research Group, Tilburg University, 2001.

Dave, K., Lawrence, S., and Pennock, D. M. Mining the Peanut Gallery: Opinion Extraction and Semantic Classification of Product Reviews. In *Proceedings of the International Conference on World Wide Web*. New York, USA, pp. 519–528, 2003.

Esuli, A. and Sebastiani, F. SentiWordNet: A Publicly Available Lexical Resource for Opinion Mining. In *Proceedings of the International Conference on Language Resources and Evaluation*. Genoa, Italy, pp. 417–422, 2006.

Fellbaum, C. WordNet - An Electronic Lexical Database. *Computational Linguistics* 25 (2): 292–296, 1998.

Hatzivassiloglou, V. and McKeown, K. R. Predicting the Semantic Orientation of Adjectives. In *Proceedings of the European Conference of the Association for Computational Linguistics*. Morristown, USA, pp. 174–181, 1997.

He, B., Patel, M., Zhang, Z., and Chang, K. C.-C. Accessing the Deep Web. *Communications of the ACM* 50 (5): 94–101, 2007.

Joachims, T. Making Large-scale SVM Learning Practical. In B. Scholkopf, C. Burgess, and A. Smola (Eds.), *Advances in Kernel Methods - Support Vector Learning*. MIT Press, Cambridge, MA, pp. 169–184, 1999.

Liu, B. Sentiment Analysis and Subjectivity. In N. Indurkhya and F. J. Damerau (Eds.), *Handbook of Natural Language Processing*. CRC Press LLC, pp. 1–38, 2010.

Nasukawa, T. and Yi, J. Sentiment Analysis: Capturing Favorability Using Natural Language Processing. In *Proceedings of the International Conference on Knowledge Capture*. New York, USA, pp. 70–77, 2003.

Nicholls, C. and Song, F. Improving Sentiment Analysis with Part-of-Speech Weighting. In *Proceedings of the International Conference on Machine Learning and Cybernetics*. Baoding, China, pp. 1592–1597, 2009.

Pang, B. and Lee, L. A Sentimental Education: Sentiment Analysis Using Subjectivity Summarization Based on Minimum Cuts. In *Proceedings of the Annual Meeting on Association for Computational Linguistics*. Morristown, USA, pp. 271–278, 2004.

Pang, B. and Lee, L. Seeing Stars: Exploiting Class Relationships for Sentiment Categorization with Respect to Rating Scales. In *Proceedings of the Annual Meeting on Association for Computational Linguistics*. Morristown, USA, pp. 115–124, 2005.

Pang, B. and Lee, L. Opinion Mining and Sentiment Analysis. *Foundations and Trends in Information Retrieval* 2 (1-2): 1–135, 2008.

Pang, B., Lee, L., and Vaithyanathan, S. Thumbs up? Sentiment Classification using Machine Learning Techniques. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Morristown, USA, pp. 79–86, 2002.

Polanyi, L. and Zaenen, A. Contextual Valence Shifters. In J. G. Shanahan, Y. Qu, and J. Wiebe (Eds.), *Computing Attitude and Affect in Text: Theory and Applications*. Springer, pp. 1–9, 2006.

Schapire, R. E. and Singer, Y. BoosTexter: A Boosting-based System for Text Categorization. *Machine Learning* 39 (2-3): 135–168, November, 2000.

Scharl, A. and Weichselbraun, A. An Automated Approach to Investigating the Online Media Coverage of US Presidential Elections. *Journal of Information Technology & Politics* 5 (1): 121–132, 2008.

Stone, P. J., Dunphy, D. C., and Smith, M. S. *The General Inquirer: A Computer Approach to Content Analysis*. M.I.T. Press, Cambridge, Massachusetts, 1966.

Subrahmanian, V. and Reforgiato, D. AVA: Adjective-Verb-Adverb Combinations for Sentiment Analysis. *Intelligent Systems, IEEE* 23 (4): 43–50, July-August, 2008.

Turney, P. D. Thumbs up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews. In *Proceedings of the Annual Meeting on Association for Computational Linguistics*. Morristown, USA, pp. 417–424, 2002.

Wiebe, J., Wilson, T., and Cardie, C. Annotating Expressions of Opinions and Emotions in Language. *Language Resources and Evaluation (formerly Computers and the Humanities)* 39 (2/3): 164–210, 2005.

Wiebe, J. M. Tracking point of view in narrative. *Computational Linguistics* 20 (2): 233–287, 1994.

Wilson, T., Wiebe, J., and Hoffmann, P. Recognizing Contextual Polarity in Phrase-level Sentiment Analysis. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*. Morristown, USA, pp. 347–354, 2005.

Wilson, T., Wiebe, J., and Hoffmann, P. Recognizing Contextual Polarity: An Exploration of Features for Phrase-Level Sentiment Analysis. *Computational Linguistics* 35 (3): 399–433, 2009.

Zdziarski, J. A. *Ending Spam: Bayesian Content Filtering and the Art of Statistical Language Classification*. No Starch Press, San Francisco, USA, 2005.