Polarity Classification Using Structure-Based Vector Representations of Text

Alexander Hogenboom, Flavius Frasincar*

Erasmus University Rotterdam, P.O. Box 1738, NL-3000 DR Rotterdam, the Netherlands

Franciska de Jong

Erasmus University Rotterdam, P.O. Box 1738, NL-3000 DR Rotterdam, the Netherlands

Universiteit Twente, P.O. Box 217, NL-7500 AE Enschede, the Netherlands

Uzay Kaymak

Eindhoven University of Technology, P.O. Box 513, NL-5600 MB Eindhoven, the Netherlands

Abstract

The exploitation of structural aspects of content is becoming increasingly popular in rule-based polarity classification systems. Such systems typically weight the sentiment conveyed by text segments in accordance with these segments' roles in the structure of a text, as identified by deep linguistic processing. Conversely, state-of-the-art machine learning polarity classifiers typically aim to exploit patterns in vector representations of texts, mostly covering the occurrence of words or word groups in these texts. However, since structural aspects of content have been shown to contain valuable information as well, we propose to use structure-based features in vector representations of text. We evaluate the usefulness of our novel features on collections of English reviews in various domains. Our experimental results suggest that, even though word-based features are indispensable to good polarity classifiers, structure-based sentiment information provides valuable additional guidance that can help significantly improve the polarity classification performance of machine learning classifiers. The most informative features capture the sentiment conveyed by specific rhetorical elements that constitute a text's core or provide crucial contextual information.

Keywords: Sentiment analysis, rhetorical structure, machine learning, support vector machines

1. Introduction

In the past decade, the Web has experienced an exponential growth into a network of more than 555 million Web sites, with over two billion users [1]. The Web has become an influential source of information with an increasing share of user-generated content, produced by many contributors [2]. This ubiquitous and ever-expanding user-generated content ranges from (micro)blog posts to reviews.

The abundance of user-generated content has the potential to act as a catalyst for well-informed decision making, as the data can be used to monitor the wants, the needs, and the opinions of large quantities of (potential) stakeholders, such as customers. Monitoring user-generated content enables decision makers to identify issues and patterns that matter, and to track and predict emerging events [3]. However, in this era of Big Data, potentially valuable data is often unstructured, scattered across the Web, and expanding at a fast rate, thus rendering manual analysis of all available data unfeasible [4]. Yet, automated tools for information monitoring and extraction can provide timely and effective support for decision making processes.

*Corresponding author; tel: +31 (0)10 408 1340; fax: +31 (0)10 408 9162

Email addresses: hogenboom@ese.eur.nl (Alexander Hogenboom), frasincar@ese.eur.nl (Flavius Frasincar), f.m.g.dejong@utwente.nl (Franciska de Jong), u.kaymak@ieee.org (Uzay Kaymak)

Today's information monitoring and extraction tools can process information from many heterogeneous sources in dynamic environments [5, 6] in order to, e.g., detect trending topics in (on-line) conversations [7], or to identify discussed entities (e.g., products or brands) and the events in which these entities play a role [8]. The past decade has brought forth a surge of research interest in extracting one type of valuable information in particular – people's sentiment with respect to entities or topics of interest [9, 10, 11, 12]. This development is driven by the significant electronic word-of-mouth effects of user-generated content [13] on, e.g., sales [14, 15] and stock ratings [16].

Many automated sentiment analysis techniques focus on determining the polarity of natural language text, typically by making use of specific cues, e.g., words, parts of words, or other (latent) features of natural language text. This is often done in machine learning methods [17, 18]. However, rule-based methods – often relying on sentiment lexicons that list words and their associated sentiment – are attractive alternatives, as the nature of typical rule-based sentiment analysis methods allows for intuitive ways of incorporating deep linguistic analysis into the sentiment analysis process [19].

Solely focusing on explicit cues for sentiment, e.g., words, has been shown not to yield a competitive polarity classification performance [20]. Therefore, successful rule-based approaches account for semantic [3] and structural [19, 21, 22, 23] aspects of content in order to improve the classification performance. Such methods typically use a text's structure in order to distinguish important text segments from less important ones in terms of their contribution to the text's overall sentiment, and subsequently weight each segment's conveyed sentiment in accordance with its identified importance.

The performance of competitive rule-based approaches, albeit comparably robust across domains and texts, is typically inferior to the performance of machine learning polarity classification systems [24]. The latter systems typically exploit patterns in (large) vector representations of texts, mainly signaling the presence of specific words or word groups in these texts. However, as structural aspects of content have been proven useful in rule-based approaches [19, 21, 22, 23], we propose to incorporate new, structure-based features in vector representations of text in order to further improve the polarity classification performance of machine learning approaches to sentiment analysis.

The main contribution of our work lies in our novel structure-based features, which facilitate a richer representation of natural language text that should enable a more accurate classification of its polarity. We evaluate the usefulness of our structure-based features in a machine learning sentiment analysis method. We thus aim to provide insight in the importance of accounting for structural aspects of text in a machine learning approach to sentiment analysis, such that automated sentiment analysis systems can be used more effectively for supporting decision making processes.

The remainder of this paper is structured as follows. First, in Section 2, we provide an introduction to the field of sentiment analysis, with a specific focus on typical features used to represent text, as well as on structure-based sentiment analysis. Then, in Section 3, we propose novel, structure-based features that can be used for sentiment analysis. We evaluate the usefulness of these features for machine learning polarity classification of text in Section 4 and we conclude in Section 5.

2. Related Work

The significant electronic word-of-mouth effects of user-generated content [13] on, e.g., sales [14, 15] and stock ratings [16] advocate a need for automated sentiment analysis methods in decision support systems [3]. With the help of such systems, organizations can pinpoint the effect of specific issues on customer perceptions, thus enabling them to respond with appropriate marketing and public relations strategies in a timely and effective manner [25]. Advances in automated sentiment analysis are hence of paramount importance for today's decision support systems.

The field of automated sentiment analysis is an upcoming field that has been attracting more and more research initiatives in the past decade [17, 18]. This surge in research interest in automated sentiment analysis techniques is fueled by the potential of sentiment analysis for real-life decision support systems [10, 26]. Several trends can be observed in existing sentiment analysis methods, as briefly addressed in Section 2.1. The vector representations of text, used by the (performance-wise) most competitive approaches are discussed in Section 2.2. In Section 2.3, we then elaborate on promising recent advances in sentiment analysis, where the analysis of the sentiment conveyed by a piece of natural language text is guided by the text's structure.

2.1. Sentiment Analysis

Existing methods for sentiment analysis focus on various tasks. Some methods deal with distinguishing subjective text segments from objective ones [27], whereas other approaches have been designed to determine the polarity of words, sentences, text segments, or documents [17]. The latter task is commonly treated as a binary classification problem, which involves classifying the polarity of a piece of text as either positive or negative. More polarity classes – e.g., classes of neutral or mixed polarity, or star ratings ranging from one to five stars – may be considered as well, yet in this paper, we address the binary classification problem for the polarity of documents. Existing binary polarity classification approaches range from rule-based to machine learning methods [17, 18].

Rule-based methods are rather intuitive methods that typically rely on sentiment lexicons, which list explicit sentiment cues like words [28] or emoticons [29], along with their sentiment scores. The scores of individual cues are typically combined in accordance with predefined rules and assumptions (e.g., by summing or averaging these scores) in order to obtain an overall sentiment score for a text, which is then used as a proxy for the text's polarity class. In this process, negation [30] or intensification [24] of sentiment may be accounted for. Moreover, rule-based sentiment analysis allows for intuitive ways of incorporating deep linguistic analysis into the process, for instance by weighting text segments in accordance with their importance, as identified based on their respective rhetorical roles [19]. The performance of rule-based methods tends to be comparably robust across domains and texts [24], and the nature of these methods allows for insight into the motivation for assigning a particular polarity class to a text.

Machine learning methods typically involve building Support Vector Machine (SVM) classifiers or the like, trained for specific corpora by means of supervised methods that aim to exploit patterns in vector representations of natural language text [24]. Such classifiers tend to yield comparably high polarity classification accuracy on the collections of texts they have been optimized for [17, 18, 24, 31], but they require a lot of (annotated) training data, as well as training time in order to reach this performance level. Nevertheless, their superior performance renders machine learning polarity classifiers particularly useful for specific, rather than generic, domain- or corpus-independent applications.

2.2. Common Features for Sentiment Analysis

Various types of features are used by existing machine learning approaches to sentiment analysis in order to construct vector representations of text. The most common and most useful features signal the presence or frequencies of specific words (i.e., unigrams) or groups of words (i.e., n-grams) [17]. Such features constitute a so-called *bag-of-words* vector representation of a text, which in itself has been shown to be rather effective in polarity classification [32, 33]. Binary features that indicate word presence have been shown to outperform frequency-based features [32], which may indicate that a text's sentiment, as opposed to its topic, is not necessarily highlighted through repeated use of the same terms [17]. Nevertheless, frequency-based features have been shown to be useful in later work [34].

Another type of information captured by features for sentiment analysis is part-of-speech (POS) information, enabling the distinction between (types of) nouns, verbs, adjectives, and adverbs. The correlation between the subjectivity of a piece of text and the presence of adjectives in this text [35] has been mistakenly taken as evidence of adjectives being good indicators for sentiment [17], thus resulting in a possibly misplaced focus on using adjectives as features in the sentiment analysis process [36, 37, 38]. Other POS types may contribute to sentiment expression too [17]. As such, a more fruitful approach is to differentiate words in the *bag-of-words* representation of a text by their POS [18].

As subjectivity is associated with word meanings rather than lexical representations of words [39, 40, 41], it is important to account for semantics when performing sentiment analysis [3]. POS information can be useful here to a limited extent [42], yet more advanced methods involve accounting for semantics by grouping words with similar meanings [38, 43].

Opinion-conveying texts are significantly different from objective texts in terms of the presence of sentiment-carrying words [44]. Specific sentiment-carrying words have therefore been used as features in so-called *bag-of-sentiwords* vector representations of text, capturing the presence of sentiment-carrying words derived from a sentiment lexicon [20, 45]. In other work, text has been represented as a *bag-of-opinions*, where features denote occurrences of unique combinations of opinion-conveying words, amplifiers, and negators [46]. Other features capture the length of a text segment, and the extent to which it conveys opinions [2, 20].

2.3. Structure-Based Sentiment Analysis

Features that capture structural aspects of content have yet to be proposed. Deep linguistic analysis can, nevertheless, help dealing with the way in which the semantic orientation of text is determined by the combined semantic orientations of its constituent phrases [47]. This compositionality can be captured by accounting for the cohesion [22] or discursive structure [19, 21, 23, 48, 49, 50] of text in the sentiment analysis process. Such structure-based sentiment analysis methods typically use a text's structure in order to distinguish important text segments from less important ones and subsequently weight each segment's conveyed sentiment in accordance with its assigned importance.

Recent advances in rule-based sentiment analysis suggest that a text's rhetorical structure, as identified by applying the Rhetorical Structure Theory (RST) [51], can be successfully exploited in order to improve polarity classification performance [19, 21, 23]. RST is a popular framework for discourse analysis. The RST framework can be used to split a piece of natural language text into segments that are rhetorically related to one another. Each segment may in turn be split as well. This process yields a hierarchical rhetorical structure, i.e., an RST tree, for the analyzed piece of text. Each segment in this tree is either a nucleus or a satellite. Nuclei form the core of a text, whereas satellites support the nuclei and are considered to be less important for understanding a text. Several types of relations exist between RST elements. A satellite may, e.g., elaborate on or form a contrast with matters presented in a nucleus. A better understanding of a text's conveyed sentiment can be obtained by differentiating between text segments, based on such rhetorical roles [19].

3. Classifying Polarity with Structure-Based Vector Representations of Text

As rule-based polarity classification has been shown to benefit from structure-guided sentiment analyses [19, 21, 23], we propose to harvest information from structural aspects of content in order to further improve an alternative, machine learning approach to polarity classification. To this end, we propose to classify the polarity of natural language text by using vector representations of text that incorporate not only word-based and sentiment-related features, but also structure-based features. Linguistic processing of a text is required in order to be able to characterize it by means of such features.

3.1. Linguistic Processing

Our framework, visualized in Figure 1, takes several steps in order to enable the extraction of features that can be used by a machine learning classifier in order to classify the polarity of a document. First, we split a document into paragraphs and, subsequently, sentences and words. Then, for each sentence, we determine the Part-of-Speech (POS) and lemma of each word. Based on the identified POS and lemma, the word sense of each word is subsequently disambiguated by means of an algorithm that iteratively selects the word sense with the highest semantic similarity to the word's context [19]. In this word sense disambiguation process, we link the identified word senses to a semantic lexical resource, i.e., WordNet [52]. WordNet is organized into sets of cognitive synonyms – synsets – which can be differentiated based on their POS type. Each out of 117,659 synsets in WordNet expresses a distinct concept and may be linked to other synsets through various types of relations, e.g., synonymy or antonymy.

Having completed these preprocessing steps, we analyze the sentiment conveyed by the document's words, given their respective POS, lemma, and sense. To this end, we retrieve the sentiment score associated with each word's POS, lemma, and word sense from a sentiment lexicon, i.e., SentiWordNet 3.0 [28], which contains positivity, negativity, and objectivity scores for each synset in WordNet. We use this information to compute sentiment scores for each word by subtracting its associated negativity score from its associated positivity score, thus yielding a real number in the interval $[-1, 1]$, representing sentiment scores in the range from very negative to very positive, respectively.

In our analysis of the sentiment conveyed by a document's words, we assign a weight to each word. These weights default to 1, but can be updated if the sentiment associated with specific words is detected to be negated or amplified. Following recent findings [30], we account for negation by inverting the polarity of the two words following a negation keyword that is listed in an existing negation lexicon [30], by multiplying their associated weights with -1. We account for amplification by means of an existing amplification lexicon [24], listing amplification keywords and their effect on the sentiment conveyed by the first succeeding word.

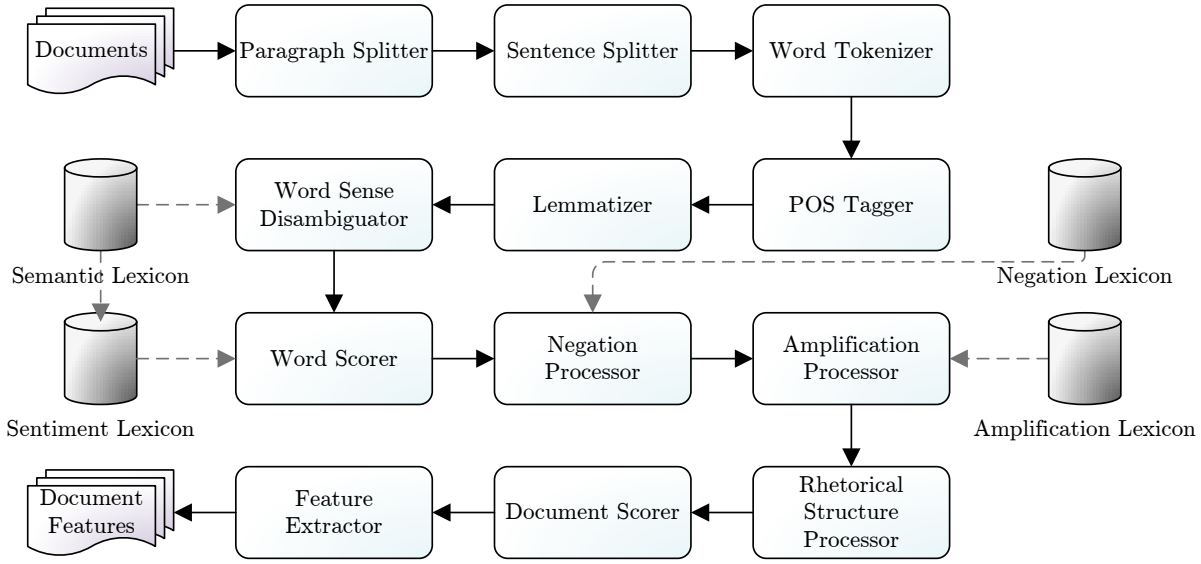

Figure 1: Overview of our sentiment analysis feature extraction framework. Solid arrows signal the information flow, whereas dashed arrows indicate a used-by relationship.

One of the final steps in our feature extraction framework involves identifying text segments and their respective rhetorical roles. In order to achieve this, we follow existing work [19, 21, 23] by segmenting the document's text in accordance with the top-level splits of sentence-level RST trees as generated by means of the SPADE parser using lexical and syntactical features [53]. Furthermore, we allow for the most fine-grained analysis of the text by performing an additional segmentation in accordance with the leaf-level splits of the sentence-level RST trees generated by the SPADE parser.

The information thus obtained can subsequently be used in order to quantify the sentiment conveyed by (parts of) a document d. We define the sentiment score ζ_{s_i} of a segment s_i as the sum of the sentiment ζ_{t_j} associated with each word t_j in segment s_i, weighted with a weight w_{t_j} associated with these respective words, i.e.,

$$\zeta_{s_i} = \sum_{t_j \in s_i} \left(\zeta_{t_j} \cdot w_{t_j}\right), \quad \forall s_i \in R_d, \tag{1}$$

with R_d representing either all top-level or all leaf-level RST nodes in the sentence-level RST trees for document d, in case of top-level or leaf-level RST-guided sentiment analysis, respectively.

The segment-level scores thus computed can subsequently be aggregated in a document-level sentiment score ζ_d, i.e.,

$$\zeta_d = \sum_{s_i \in R_d} \zeta_{s_i}. \tag{2}$$

Once a document's sentiment score has been computed, features can be extracted in order to characterize the document in a way that allows for its polarity to be determined.

3.2. Extracted Features

As discussed in Section 2.2, common, valuable features to be included in a vector representation of text capture the (frequencies of) occurrence of specific words. These words could be simple lexical representations (i.e., strings of characters), or more complex ones, e.g., WordNet synsets. Inspired by the state-of-the-art (see Section 2.2), we use both representations. First, we represent text by means of the WordNet synsets (unigrams) that can be identified in its contents, as these synsets capture semantics and can be differentiated by their POS. Second, we represent text by means of its constituent lemmas (unigrams and bigrams), differentiated by their POS, in order to cover words that do not have an entry in WordNet. Another extracted word-based feature is the length of a document, expressed in terms of its total number of words, as a text segment's length has been shown to be a potentially useful feature in sentiment analysis, too.

Other common, useful features relate to the sentiment conveyed by the text, as determined by means of a sentiment lexicon (see Section 2.2). Therefore, our framework extracts sentiment-related features that include the number of positive words, the number of negative words, and the sentiment scores of sentiment-carrying words, aggregated by means of (1) for segments and (2) for documents. As information on negation and amplification is valuable when representing sentiment-carrying content in vector representations of text, we construct our sentiment-related features when performing four distinct types of sentiment analysis. We construct our sentiment-related word counts and scores when performing sentiment analysis without accounting for negation and amplification, sentiment analysis accounting for negation, sentiment analysis accounting for amplification, and sentiment analysis accounting for negation and amplification.

The sentiment-related features extracted by our framework can be used to characterize documents as a whole, but we propose to apply them to each distinct type of rhetorical element as well. Here, we define a rhetorical element as a text segment that has been identified as a nucleus or satellite belonging to a type of rhetorical relation on a specific level of analysis. A rhetorical element may for instance be an attributing satellite or the nucleus of a contrasting relation, in either the top-level split or a leaf-level split of a sentence-level RST tree. Our framework constructs features that capture the total number of words, the number of positive and negative words, and aggregated sentiment scores of the text segments that have been identified as specific rhetorical elements. The element-level features thus constructed allow for words and their conveyed sentiment to be treated differently, depending on their identified rhetorical role.

4. Evaluation

We evaluate the usefulness of our proposed (structure-based) features for polarity classification in a set of experiments. The setup of these experiments is detailed in Section 4.1. Additionally, we present our experimental results and discuss some caveats with respect to our findings in Section 4.2.

4.1. Experimental Setup

We evaluate the performance of our features in a binary polarity classification task on two collections of documents. The first collection consists of 1,000 positive and 1,000 negative English movie reviews [33]. The second corpus is a multi-domain collection of 8,000 English reviews, consisting of 1,000 positive and 1,000 negative reviews for each out of four distinct product categories, i.e., books, DVDs, electronics, and kitchen appliances [54].

4.1.1. Implementation

Feature extraction is performed by means of a Java-based implementation of our proposed framework. The initial tokenization steps in this implementation vary for our corpora. For the movie review data, we detect paragraphs by making use of the `<P>` and `</P>` tags in the original HTML files of the reviews, as these tags signal the respective starts and ends of paragraphs. In order to segment the identified paragraphs into sentences, we rely on the preprocessing done by Pang and Lee [33]. Conversely, for the multi-domain review corpus, we detect paragraphs by considering white lines to separate paragraphs. The paragraphs thus identified are split into sentences by means of the Stanford CoreNLP toolkit [55]. Then, for both corpora, we identify words using the Stanford Tokenizer [56].

In order to identify the POS and lemma of each word thus identified, we use the OpenNLP [57] POS tagger and the Java WordNet Library (JWNL) API [58], respectively. Only those words occurring in WordNet are actually lemmatized, whereas the lemma of each other word is in fact its original form. We link the words' senses to WordNet [52] and retrieve their sentiment scores from SentiWordNet 3.0 [28]. Using a negation lexicon [30], we then invert the polarity of the two words following negation keywords. We account for amplification by means of a lexicon that lists amplification keywords and their effect on the sentiment conveyed by the first succeeding word [24]. Last, we identify the rhetorical roles of words by analyzing the top-level and leaf-level splits of sentence-level RST trees, generated by SPADE [53].

4.1.2. Experiments

We consider feature sets in three categories, i.e., four sets of word-based features, one set of sentiment-related features, and two sets of RST-based features (see Table 1). We assess the merits of each set individually, as well as in combination with other sets, with each combination containing at most one set from each category. Evaluating the performance of these combinations helps us assess the added value of each individual set of features.

Set	Type	Description
$\mathcal{B}$	Words	Document-level, binary features that indicate the presence of synset unigrams.
$\mathcal{F}$	Words	Document-level features indicating the frequencies of occurrence of synset unigrams.
$\mathcal{N}$	Words	Document-level, binary features that indicate the presence of lemma-based n-grams, i.e., unigrams and bigrams that differentiate lemmas by POS.
$\mathcal{W}$	Words	Document-level features indicating the frequencies of occurrence of lemma-based n-grams, i.e., unigrams and bigrams that differentiate lemmas by POS.
$\mathcal{S}$	Sentiment	Document-level features capturing the sentiment scores and the total, positive, and negative word counts, for four types of sentiment analysis.
$\mathcal{T}$	RST	The sentiment scores and the total, positive, and negative word counts, for four types of sentiment analysis, differentiated by top-level RST element type.
$\mathcal{L}$	RST	The sentiment scores and the total, positive, and negative word counts, for four types of sentiment analysis, differentiated by leaf-level RST element type.

Table 1: Feature sets used in our experiments.

The word-based sets $\mathcal{B}$ and $\mathcal{F}$ contain features that indicate the respective presence and frequencies of occurrence of all WordNet synsets that occur in at least 5% of our data, i.e., 997 synsets for the movie review corpus, and 322 synsets for the multi-domain corpus. We apply this filter in order to keep the number of features tractable – considering all WordNet synsets would result in 117,659 features. Moreover, even though rare terms may be useful indicators for subjectivity [27], excluding such terms can yield models that generalize comparably well.

Similarly, the word-based feature sets $\mathcal{N}$ and $\mathcal{W}$ encompass features indicating the respective presence and frequencies of occurrence of all POS-specific lemma unigrams and bigrams that occur in at least 5% of our data, i.e., 1,157 n-grams for the movie review corpus, and 388 n-grams for the multi-domain corpus. This vastly reduces the feature space of 524,855 and 425,320 initially extracted n-grams for the movie review corpus and multi-domain review corpus, respectively.

Set $\mathcal{S}$ contains 16 features that capture the sentiment conveyed by a review's full text. These features represent the sentiment score and the total, positive, and negative word counts, as obtained by performing document-level sentiment analysis without accounting for negation and amplification (SA), sentiment analysis accounting for negation (SA^{-}), sentiment analysis accounting for amplification (SA^{+}), and sentiment analysis accounting for negation and amplification ($SA^{\pm}$).

The RST-based feature sets $\mathcal{T}$ and $\mathcal{L}$ each contain 480 features representing 16 sentiment-related concepts for rhetorical elements in top-level ($\mathcal{T}$) or leaf-level ($\mathcal{L}$) splits of sentence-level RST trees. They encompass the nucleus and satellite elements for 14 rhetorical relations (see Table 2) that occur in at least 5% of our data (thus dealing with data sparsity), as well as a nucleus and a satellite element representing all other nuclei and satellites.

We assess the performance of each of our (combined) feature sets in terms of the F_1-score for positive and negative documents separately, as well as the macro-level F_1-score, i.e., the arithmetic mean of the F_1-scores for the positive and negative documents, weighted for their relative frequencies. The F_1-score is the harmonic mean of the disparate measures of precision and recall, thus rendering it a useful overall statistic. Precision is the proportion of the positively (negatively) classified documents that are in fact positive (negative), whereas recall is the proportion of the actual positive (negative) documents that are also classified as such. We assess the significance of performance differences by means of a paired two-sample two-tailed t-test.

The performance is assessed under 10-fold cross-validation. For the movie review data as well as for each domain in the multi-domain review corpus, we randomly split the data into ten balanced folds, with 100 positive and 100 negative reviews each. For each (combined) set of features, our evaluation procedure is as follows. For each fold, we select features on the fold's training data (see Section 4.1.3). A polarity classifier that uses the selected features is then trained on the training data, and we evaluate its document polarity classification performance on the fold's test data (see Section 4.1.4). For each corpus, the resulting performance measures are subsequently aggregated over all folds in order to assess the overall performance of our feature sets.

Relation	Satellite description
Attribution	Clause containing reporting verbs or cognitive predicates related to reported messages presented in the nucleus.
Background	Information helping a reader to sufficiently comprehend matters in the nucleus.
Cause	An event leading to a result presented in the nucleus.
Comparison	Examination of matters along with matters presented in the nucleus.
Condition	Hypothetical, future, or otherwise unrealized situations, the realization of which influences the realization of nucleus matters.
Contrast	Situations juxtaposed to and compared with situations in the nucleus, which are considered as mostly similar, yet different in a few respects.
Elaboration	Additional detail about matters presented in the nucleus.
Enablement	Information increasing a reader's potential ability of performing actions presented in the nucleus.
Evaluation	Evaluative comments about matters presented in the nucleus.
Explanation	Justifications or reasons for situations presented in the nucleus.
Joint	No specific rhetorical relation holds with the matters presented in the nucleus.
Manner-Means	Explains how or by which means matters presented in the nucleus have been done.
Same-Unit	Text segment of which the subordinate nucleus belongs to the same rhetorical unit as the nucleus.
Temporal	Events with an ordering in time with respect to events in the nucleus.

Table 2: Most common relations of satellites to their nuclei, as identified by SPADE in our data.

4.1.3. Feature Selection

When performing feature selection on a training set, we first remove the features that show no variation over the training instances, as these features contain no information that can be used to distinguish between positive and negative polarity. Then, we rank the remaining features by the absolute value of their (Pearson) correlation with the document polarity and select those features with an absolute Pearson correlation coefficient of 0.1 or higher, in order to keep only those features that are at least somewhat relevant.

The absolute value of the Pearson correlation coefficient is a widely used ranking criterion, which is applicable to binary, continuous, and even (disjunctively coded) categorical features and target variables [59]. Our considered features are both binary and continuous, whereas our target variable, i.e., document polarity, is a categorical variable. As such, the absolute Pearson correlation coefficient is an attractive feature selection criterion for our data. The alternative wrapper methods for feature selection are less suitable in our particular case, due to the inherent computational complexity involved with evaluating the performance of the combinatorial explosion of subsets of features that can be constructed from our feature sets.

4.1.4. Polarity Classification

Using only those features selected by means of the procedure described in Section 4.1.3, we train a machine learning classifier on a training set and evaluate its polarity classification performance on a test set. In this work, we use an SVM classifier, as such classifiers are often used in polarity classification tasks [24]. We use the WEKA [60] implementation of an SVM classifier, i.e., the *SMO* classifier, with a Radial Basis Function (RBF) kernel.

Two parameters of this classifier can be optimized, i.e., the parameters γ and C, both of which capture a trade-off between the complexity of the decision surface and the misclassification of training instances. A decision surface that is too complex may result in overfitting, so optimizing these parameters is of paramount importance. Therefore, we optimize γ and C on the training data.

Our three-step parameter optimization procedure aims to find the values for γ and C that give the best accuracy on the training set, as assessed by means of internal 10-fold cross-validation. In the first step of our procedure, we perform a grid search on a logarithmic grid with base 10, with values of $\{10^{-3}, 10^{-2}, \ldots, 10^{3}\}$ for both γ and C. Then, we perform an additional grid search on a logarithmic grid with base 1.5, between the grid points surrounding the optimum found in the first iteration.

	Movies			Multi-domain		
Features	Positive	Negative	Overall	Positive	Negative	Overall
$\mathcal{B}$	0.774	0.777	0.775	0.714	0.715	0.714
$\mathcal{F}$	0.765	0.772	0.769	0.726	0.715	0.720
$\mathcal{N}$	0.785	0.784	0.784	0.736	0.732	0.734
$\mathcal{W}$	0.807	0.810	0.808	0.748	0.740	0.744
$\mathcal{S}$	0.623	0.667	0.645	0.705	0.700	0.702
$\mathcal{T}$	0.657	0.674	0.665	0.709	0.708	0.708
$\mathcal{L}$	0.645	0.669	0.657	0.708	0.706	0.707
$\mathcal{BS}$	0.780	0.781	0.780	0.757	0.756	0.756
$\mathcal{BT}$	0.791	0.792	0.791	0.767	0.766	0.766
$\mathcal{BL}$	0.782	0.782	0.782	0.758	0.755	0.756
$\mathcal{FS}$	0.781	0.786	0.783	0.759	0.758	0.758
$\mathcal{FT}$	0.785	0.791	0.788	0.762	0.764	0.763
$\mathcal{FL}$	0.780	0.785	0.782	0.761	0.764	0.763
$\mathcal{NS}$	0.806	0.803	0.804	0.765	0.760	0.762
$\mathcal{NT}$	0.802	0.805	0.803	0.772	0.771	0.771
$\mathcal{NL}$	0.807	0.802	0.804	0.771	0.770	0.770
$\mathcal{WS}$	0.814	0.816	0.815	0.780	0.773	0.777
$\mathcal{WT}$	**0.819**	**0.820**	**0.819**	0.781	0.778	0.779
$\mathcal{WL}$	0.817	0.820	0.818	0.777	0.774	0.776
$\mathcal{ST}$	0.658	0.684	0.671	0.713	0.713	0.713
$\mathcal{SL}$	0.663	0.680	0.672	0.708	0.705	0.706
$\mathcal{BST}$	0.791	0.791	0.791	0.766	0.765	0.766
$\mathcal{BSL}$	0.780	0.781	0.780	0.764	0.763	0.763
$\mathcal{FST}$	0.782	0.788	0.785	0.768	0.769	0.769
$\mathcal{FSL}$	0.781	0.784	0.782	0.763	0.765	0.764
$\mathcal{NST}$	0.805	0.807	0.806	0.775	0.774	0.775
$\mathcal{NSL}$	0.812	0.810	0.811	0.770	0.769	0.769
$\mathcal{WST}$	0.815	0.819	0.817	**0.783**	**0.780**	**0.781**
$\mathcal{WSL}$	0.814	0.820	0.817	0.779	0.775	0.777

Table 3: The 10-fold cross-validated F_1-scores of our feature sets on the movie review corpus and the multi-domain review corpus. The best performance is printed in bold for each performance measure.

Last, we perform a grid search between the grid points around the optimum found in the second iteration, on a logarithmic grid with base 1.05.

After having optimized the γ and C parameters of our SVM classifier, we train the classifier on the full training set, using the optimized parameters. We then evaluate the polarity classification performance of the classifier on the test set.

4.2. Experimental Results

The machine learning classifiers that use our various sets of features exhibit several trends in terms of polarity classification performance, as discussed in Section 4.2.1. The features selected by our machine learning polarity classifiers are analyzed in Section 4.2.2. Some caveats with respect to our findings our discussed in Section 4.2.3.

4.2.1. Polarity Classification Performance

The various combinations of features result in the polarity classification performance statistics reported in Table 3 and Figures 2 and 3. The precision and recall scores constituting the reported F_1-scores are rather well-balanced, even though our classifiers tend to have a slightly higher precision on positive documents, and a higher recall on negative documents. Furthermore, our classifiers' performance exhibits a rather large variation over the feature sets. On the movie review corpus, the 10-fold cross-validated macro-level F_1-scores range from about 65% for the worst-performing classifiers to approximately 82% for the best-performing ones. The macro-level F_1-scores on the multi-domain review data range from about 70% to 78%.

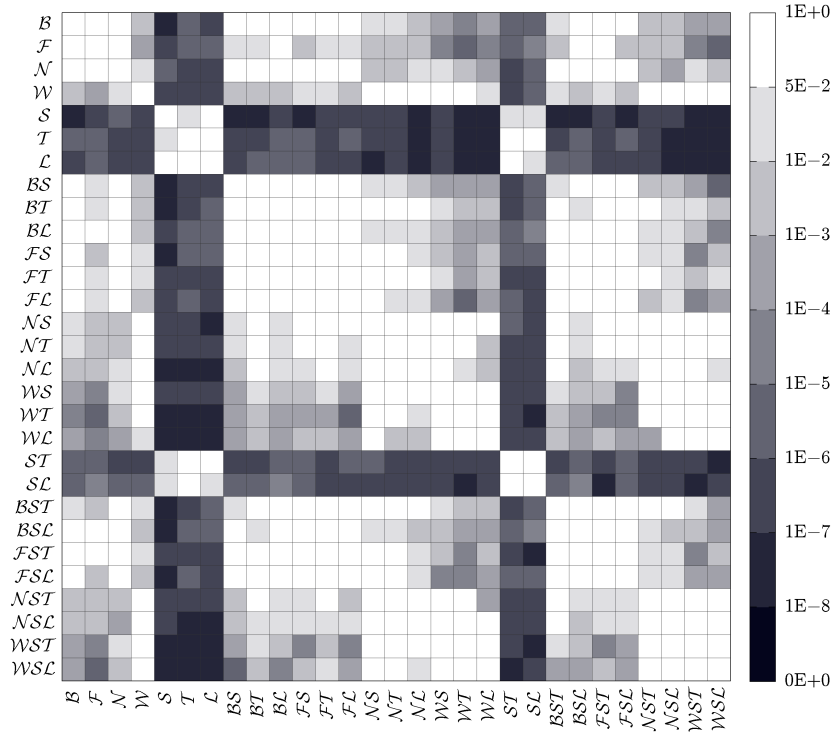

Figure 2: The p-values for the paired, two-tailed t-test assessing the statistical significance of differences in mean macro-level F_1-scores obtained by using our (combined) feature sets on the movie review corpus.

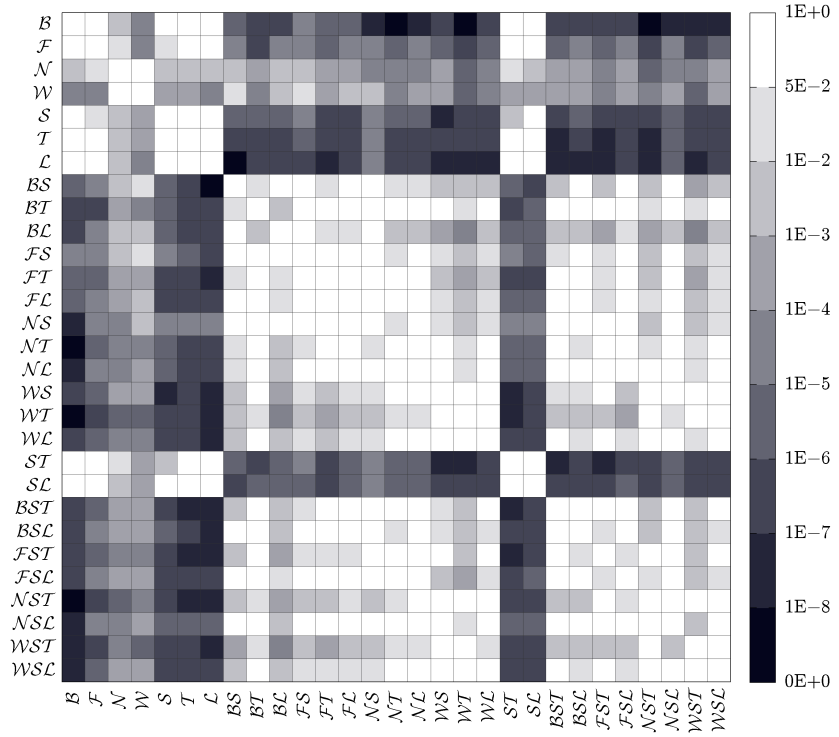

Figure 3: The p-values for the paired, two-tailed t-test assessing the statistical significance of differences in mean macro-level F_1-scores obtained by using our (combined) feature sets on the multi-domain review corpus.

	Movies				Multi-domain			
Features	$+\mathcal{B}$	$+\mathcal{F}$	$+\mathcal{N}$	$+\mathcal{W}$	$+\mathcal{B}$	$+\mathcal{F}$	$+\mathcal{N}$	$+\mathcal{W}$
$\mathcal{S}$	0.210	0.214	0.247	0.263	0.077	0.080	0.086	0.106
$\mathcal{T}$	0.190	0.184	0.208	0.232	0.082	0.077	0.089	0.100
$\mathcal{L}$	0.191	0.192	0.225	0.247	0.069	0.078	0.089	0.097
$\mathcal{ST}$	0.179	0.170	0.201	0.218	0.074	0.078	0.087	0.096
$\mathcal{SL}$	0.162	0.165	0.207	0.216	0.081	0.081	0.089	0.100

Table 4: Relative change in the 10-fold cross-validated macro-level F_1-scores on the movie review corpus and the multi-domain review corpus when including word-based features. All performance differences are statistically significant at $p < 0.0001$.

The worst-performing classifiers use (combinations of) the sentiment-based features in $\mathcal{S}$, the leaf-level RST-based features in $\mathcal{L}$, and the top-level RST-based features in $\mathcal{T}$. However, these features become particularly useful once combined with the comparably well-performing word-based features in $\mathcal{B}$, $\mathcal{F}$, $\mathcal{N}$, and especially $\mathcal{W}$. Combinations of feature sets typically yield a better overall performance than each feature set individually. Our best classifiers include (mostly top-level) RST-based features, sometimes combined with document-level sentiment-related features. Our three best movie review classifiers use feature set combinations $\mathcal{WT}$, $\mathcal{WL}$, and $\mathcal{WST}$, and the best three classifiers for the multi-domain review corpus use feature set combinations $\mathcal{WST}$, $\mathcal{WT}$, and $\mathcal{WSL}$.

Our results reveal subtle performance differences between similar feature sets. For instance, feature sets that include frequency-based word features from $\mathcal{F}$ and $\mathcal{W}$ more often than not outperform their respective binary counterparts $\mathcal{B}$ and $\mathcal{N}$. However, these differences are mostly statistically insignificant. Similarly, top-level RST-based features in $\mathcal{T}$ appear to be associated with a better overall polarity classification performance than leaf-level RST-based features in $\mathcal{L}$, but these performance differences are not statistically significant either. On the other hand, lemma-based features from sets $\mathcal{N}$ and $\mathcal{W}$ tend to yield significantly better polarity classification performance than synset-based features from sets $\mathcal{B}$ and $\mathcal{F}$, especially on the movie review corpus. Because the general purpose WordNet synsets do not cover all words occurring in the reviews, our lemma-based features can represent the reviews' content more accurately, thus facilitating a more accurate polarity classification.

In general, individual feature sets, i.e., $\mathcal{B}$, $\mathcal{F}$, $\mathcal{N}$, $\mathcal{W}$, $\mathcal{S}$, $\mathcal{T}$, and $\mathcal{L}$, tend to perform better once they are combined with one another. The classifiers that use features from multiple feature sets exhibit the best performance in our experiments. Tables 4, 5, and 6 provide insight into the effects of combining word-based, sentiment-related, and RST-based features, respectively, with one another.

	Movies	Multi-domain
Features	$+\mathcal{S}$	$+\mathcal{S}$
$\mathcal{B}$	0.006	0.058***
$\mathcal{F}$	0.019**	0.053***
$\mathcal{N}$	0.025**	0.039***
$\mathcal{W}$	0.008	0.044***
$\mathcal{T}$	0.009	0.006
$\mathcal{L}$	0.023*	-0.001
$\mathcal{BT}$	-0.001	-0.001
$\mathcal{BL}$	-0.002	0.009**
$\mathcal{FT}$	-0.004	0.008*
$\mathcal{FL}$	0.000	0.002
$\mathcal{NT}$	0.003	0.005*
$\mathcal{NL}$	0.008	-0.001
$\mathcal{WT}$	-0.003	0.003
$\mathcal{WL}$	-0.002	0.002

Table 5: Relative change in the 10-fold cross-validated macro-level F_1-scores on the movie review corpus and the multi-domain review corpus when including sentiment-related features. Performance differences marked with * are statistically significant at $p < 0.05$, those marked with ** are significant at $p < 0.01$, and those marked with *** are significant at $p < 0.001$.

Table 4 clearly shows that adding word-based features from sets $\mathcal{B}$, $\mathcal{F}$, $\mathcal{N}$, or $\mathcal{W}$ to sentiment-related or RST-based features yields vast, significant performance improvements. This confirms the substantial importance of word-based features for polarity classification purposes, as already suggested in work discussed in Section 2.2. For the movie review corpus, the performance improvements obtained by adding synset-based features or especially lemma-based features to sets $\mathcal{S}$, $\mathcal{L}$, and $\mathcal{T}$ amount to about 26%, yet they drop to about 20% when adding these features to combinations of these sets.

	Movies		Multi-domain	
Features	$+\mathcal{T}$	$+\mathcal{L}$	$+\mathcal{T}$	$+\mathcal{L}$
$\mathcal{B}$	0.021	0.008	0.073^{***}	0.058^{***}
$\mathcal{F}$	0.025^{*}	0.018^{*}	0.059^{***}	0.059^{***}
$\mathcal{N}$	0.024^{**}	0.025^{*}	0.051^{***}	0.050^{***}
$\mathcal{W}$	0.014	0.012^{*}	0.048^{***}	0.043^{***}
$\mathcal{S}$	0.040^{*}	0.041^{*}	0.015^{**}	0.006
$\mathcal{BS}$	0.013^{*}	0.000	0.012^{**}	0.009
$\mathcal{FS}$	0.002	-0.001	0.014^{*}	0.007
$\mathcal{NS}$	0.002	0.008	0.016^{**}	0.009
$\mathcal{WS}$	0.002	0.002	0.006	0.000

Table 6: Relative change in the 10-fold cross-validated macro-level F_1-scores on the movie review corpus and the multi-domain review corpus when including RST-based features. Performance differences marked with * are statistically significant at $p < 0.05$, those marked with ** are significant at $p < 0.01$, and those marked with *** are significant at $p < 0.001$.

This pattern is less clear-cut for the multi-domain review corpus, where adding word-based features generally yields overall performance improvements ranging from about 7% to 10%.

The added value of the document-level sentiment-related features in $\mathcal{S}$ is more limited, as exhibited by Table 5. Adding sentiment-related features to word-based or RST-based features can yield modest overall performance improvements of up to about 2% and 6% on the movie review corpus and the multi-domain corpus, respectively. These performance improvements are mostly statistically significant for word-based features and – on the movie review corpus – for (leaf-level) RST-based features. Adding sentiment-related features from $\mathcal{S}$ to combined word-based and RST-based features does not yield substantial performance improvements. This suggests that the document-level sentiment-related information in feature set $\mathcal{S}$ does not add much to the information that is already covered by the well-performing combinations of word-based features with our novel RST-based features that capture sentiment-related information on the level of rhetorical elements.

Adding RST-based features to word-based features or document-level sentiment-related features yields mostly significant, yet typically modest improvements in overall polarity classification performance of up to approximately 4% on the collection of movie reviews, and over 7% on the multi-domain review corpus (see Table 6). The RST-based sentiment-related information in feature sets $\mathcal{T}$ and $\mathcal{L}$ has the most convincing added value over the information represented by individual sets of word-based features or document-level sentiment-related features, i.e., sets $\mathcal{B}$, $\mathcal{F}$, $\mathcal{N}$, $\mathcal{W}$, and $\mathcal{S}$. Smaller improvements in polarity classification performance (if any at all) can be achieved by adding RST-based features to combined word-based and document-level sentiment-related features. The latter performance improvements are only statistically significant on the multi-domain review corpus, whereas the performance improvements realized on individual feature sets tend to be statistically significant across our considered corpora. On the movie review corpus, sets $\mathcal{B}$ and $\mathcal{W}$ form the only exception to this observation. Yet, the 2% overall performance improvement obtained by adding feature set $\mathcal{T}$ to the binary synset-based feature set $\mathcal{B}$ is just short of qualifying as statistically significant on the movie review corpus, with a p-value of 0.050.

All in all, the inclusion of word-based features in our machine learning polarity classifier seems to have the most impact on the overall polarity classification performance on our considered corpora. However, adding sentiment-related information, especially on the level of rhetorical elements, can yield modest, yet significant performance improvements as well – models that include such information generally significantly outperform their counterparts that do not include such information.

4.2.2. Selected Features

The polarity classification performance reported in Section 4.2.1 is in fact obtained by using comparably small subsets of features, that have been selected by means of the feature selection procedure described in Section 4.1.3. On average, only about 7% of all extracted features is actually used in our classifiers. The only exception here is our smallest feature set, i.e., $\mathcal{S}$, where on average about 70% of all extracted features is selected.

Our comparably well-performing classifiers generally use more features (in absolute terms) than the less competitive classifiers. Nevertheless, using more features does not guarantee a better performance. Our three best-performing movie review classifiers use on average 137, 132, and 149 features from the $\mathcal{WT}$, $\mathcal{WL}$, and $\mathcal{WST}$ sets, respectively, whereas some other classifiers perform worse on this corpus while using a similar or even higher number of features. Similarly, our three best-performing classifiers on the multi-domain review data use on average 46, 39, and 47 features from the $\mathcal{WST}$, $\mathcal{WT}$, and $\mathcal{WSL}$ sets, whereas some of the less competitive classifiers use a comparable or higher amount of features. Clearly, the quality of features is important as well. Our best-performing models' most important features – i.e., those most strongly correlated with document polarity – exhibit several patterns, as demonstrated by Figures 4, 5, and 6.

The characteristics of the single most important feature selected for each out of ten folds for the three best-performing sets of features for both considered corpora are visualized in Figure 4. In 25% of the cases, the single most important selected feature captures information related to document-level sentiment, whereas in 75% of the cases, the most important feature captures sentiment-related information on the level of rhetorical elements. Interestingly, word presence or frequencies do not turn out to be among the single most important features, in spite of their strong and significant impact on the performance of our classifiers, as discussed in Section 4.2.1. An explanation for this phenomenon lies in the comparably complex nature of our sentiment-related features, which condense a lot of information related to how specific words are used in order to convey sentiment.

Our models' most valuable document-level sentiment-related features capture lexicon-based sentiment scores. These scores stem from the method discussed in Section 3.1 and account for both negation and amplification. The most important RST-based features capture similar sentiment scores, computed for some nuclei of mostly top-level splits of sentence-level RST trees. These nuclei do not belong to the 14 most salient rhetorical relations, but capture the combined nuclei of all other rhetorical relations, and thus cover the core information for many rhetorical roles at once.

Figure 5 demonstrates the more varied nature of the ten best features that have been selected for each fold for our best-performing feature sets. Clearly, specific words used in our reviews are important indicators for the polarity of these reviews, yet sentiment-related information – especially the RST-based variant – dominates the top ten features. Document-level sentiment-related features cover 24% of the top ten selected features and RST-based sentiment-related features cover another 42% of the top ten selected features, whereas the remaining 34% consists of word-based features.

Word-based features included in the ten best features of the models that yield the best performance on our corpora are mostly frequencies of lemmas. The most useful lemmas are typically opinion-expressing adjectives, such as "*bad*" (also in combination with the noun "*movie*"), "*ridiculous*", "*good*", and "*great*". An interesting informative adverb is "*not*", sometimes preceded by the verb "*to do*" – in our data, negative opinions often tend to be expressed or even emphasized by negating the opposite. The nouns "*life*", "*money*", and "*price*" are valuable indicators for the polarity of a review as well. The high discriminative power of the word "*life*" appears to be specific to the movie and DVD review domain, where quite a few positive reviews describe how well a movie captures (the struggles, challenges, or absurdity of) real life. In the multi-domain review corpus, "*money*" is much more likely to be used in a negative context, e.g., in order to express that a product is a waste of money. Conversely, "*price*" is much more likely to be used in a positive context, e.g., in order to express that a product is attractively priced. Another rather peculiar, yet important feature turns out to be the verb "*to suppose*". Reviewers often use this verb in order to express that their expectations have not been met (e.g., "*Her side-kick was supposed to be funny but just annoyed me*"). Another informative verb is "*to waste*", which is typically used in order to express a perceived waste of money or talent. Other useful verbs are "*to enjoy*", "*to love*", and "*to return*", the latter of which is often used in a negative context, e.g., in order to express that a reviewed item was or should be returned to the store.

The document-level sentiment-related features in the top ten features of our best-performing models cover sentiment scores computed by performing sentiment analysis without accounting for negation or amplification, or by performing a type of sentiment analysis that accounts for negation, amplification, or both negation and amplification. A similar pattern can be observed for the RST-based sentiment-related features in the top ten features.

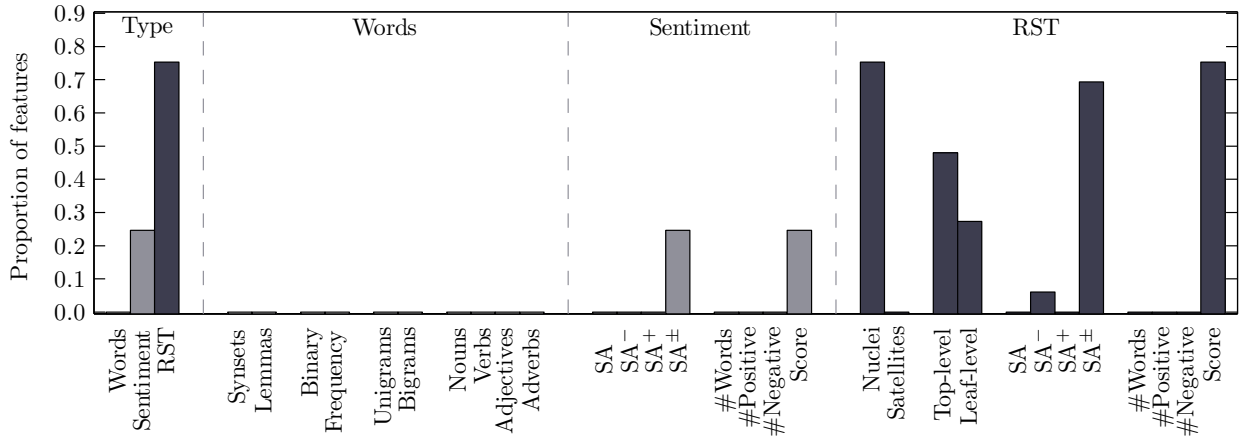

Figure 4: Characteristics of the top 1 features selected for all folds of the three best-performing feature sets for each corpus.

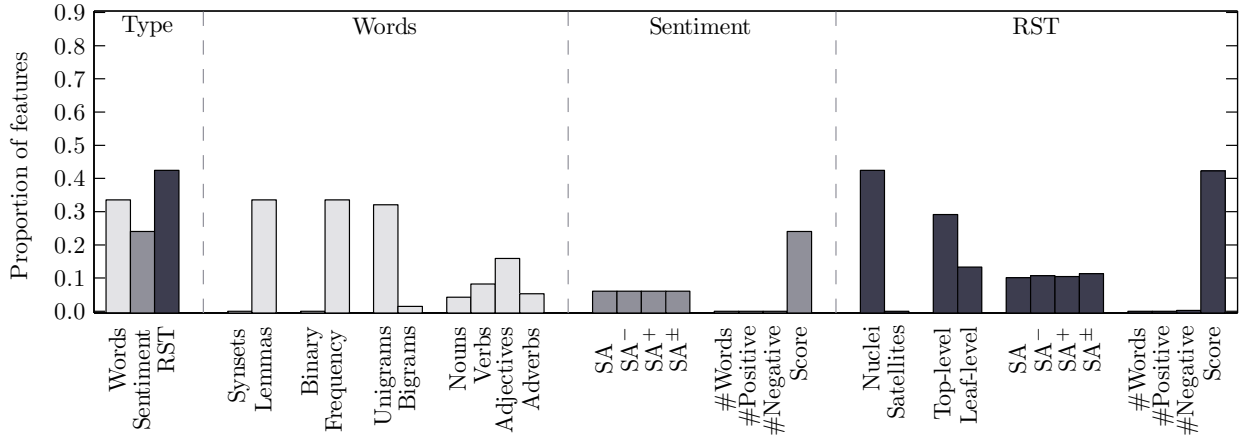

Figure 5: Characteristics of the top 10 features selected for all folds of the three best-performing feature sets for each corpus.

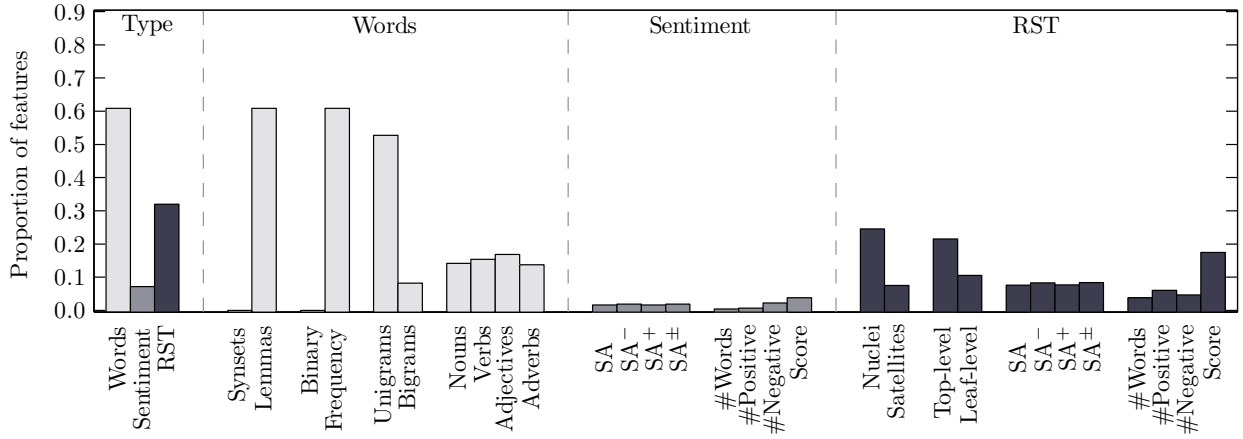

Figure 6: Characteristics of all features selected for all folds of the three best-performing feature sets for each corpus.

These features relate to (mostly top-level) nuclei only and cover – besides the nuclei covered by the single best features – the JOINT nuclei, which occur in almost every review and thus cover a substantial part of the core content of many reviews.

Figure 6 shows that even in all features selected by the models based on our best-performing feature sets, sentiment-related information is valuable, especially when this information is RST-based. Nevertheless, word-based features form a small majority of all selected features, i.e., 61%. Document-level and RST-based sentiment-related features cover another 7% and 32%, respectively.

Besides the words covered by the top ten features, the word-based features selected by our best-performing models cover the lemmas of many adjectives, adverbs, nouns, and verbs. The numerous additional adjectives include "*awful*", "*boring*", "*predictable*", "*memorable*", and "*little*". The latter adjective is typically used in terms of endearment (e.g., "*This little gem*"), or in order to downplay negative aspects of a product in an otherwise positive review (e.g., "*The soup bowls are a little on the small side*"). Additional adverbs include "*unfortunately*", "*well*", and "*instead*", the latter of which is typically used in order to express a mismatch between expectations and reality. Noteworthy additional nouns include "*nothing*" (e.g., "*Nothing in this movie makes sense*"), "*flaw*", "*performance*" (typically used in order to express that an actor delivered quite a performance), "*service*", and "*support*". The nouns "*service*" and "*support*" are especially valuable proxies for negative sentiment in the electronics domain, where needing support turns out to be a good indicator for bad product experiences. Last, noteworthy additional verbs include "*to recommend*", "*to deserve*", "*to fail*", as well as the verb "*to be*", combined with numerous positive and negative adjectives like "*great*" and "*bad*".

All sentiment-related features used in our best-performing models cover sentiment scores and – to a more limited extent – word counts. This information is typically obtained by performing any of our considered sentiment analysis variants, but preferably by means of a variant that at least accounts for negation of the sentiment conveyed by specific words. Selected RST-based sentiment-related features cover rhetorical relations in mostly top-level splits of sentence-level RST trees. Most of these features cover nuclei, but some satellites are represented too. This suggests that satellites – which are considered to contain less relevant information – in fact contain useful information that can help distinguish positive from negative texts. For instance, ELABORATION satellites, which provide additional details, and the persuasive ENABLEMENT satellites are important. Additionally, our best-performing models often include features that capture the sentiment in ATTRIBUTION satellites, which present the context of messages reported in nuclei. Attributing satellites may contain more useful information than reported messages per se, as is the case in the phrase "*Any studio executive that thinks this plot is going to win points with the reviewing press needs to check into rehab*". Here, the reported message of the plot being praised by the reviewing press is subordinate to its negative context. Another important satellite turns out to be the CONDITION satellite, which provides crucial prerequisites for matters presented in nuclei. For example, in the phrase "*We wouldn't mind a minute of Johnny Mnemonic if the action played better*", the nucleus suggests a positive sentiment with respect to the movie, whereas the satellite clarifies that this would only hold if it were not for the lousy action.

Overall, in our best-performing polarity classifiers, sentiment-carrying words – especially adjectives – turn out to be valuable features. Our best classifiers use features that capture the (frequency of) occurrence of specific lemmas (predominantly unigrams). The most valuable information, however, appears to be derived from sentiment-related, and mostly RST-based features. Especially nuclei of top-level splits of sentence-level RST trees turn out to contain valuable cues for the polarity of movie reviews, yet some types of satellites that provide crucial contextual information play an important role as well. Hence, features that capture sentiment information, especially when related to the structure of documents, form a valuable addition to commonly used word-based features.

4.2.3. Caveats

In spite of our promising results, several caveats should be taken into consideration. First, some of our word-based features are linked to the semantic categories in a general purpose semantic lexicon, i.e., to synsets in WordNet. As explained in Section 3.2, such a representation enables us to capture the semantics and POS information of words, thus allowing for more robust models. However, the WordNet synsets may not cover all lexical representations of words occurring in a corpus. Highly domain-specific words may not be covered either.

This explains why the word-based features that are based on lexical representations of (the lemmas of) words tend to yield a better polarity classification performance. The trade-off between robustness and domain-specificity may affect the quality of the document-level and RST-based sentiment-related features as well, as these features rely on the SentiWordNet 3.0 sentiment lexicon, which only contains sentiment scores for each WordNet synset.

Another caveat is related to our feature selection process. We disregard features that occur in only a small part of our corpora, even though these features could be valuable [27]. Moreover, we disregard features that are hardly correlated with the polarity class of the reviews in our corpora. This methodology can be justified as it allows us to reduce the dimensionality of our data and to make our models less prone to overfitting. However, other subsets of features may exist that yield an even better polarity classification performance than our current sets can. These alternative subsets may be found by using other feature selection methods, for instance by means of genetic algorithms or ant colony optimization techniques that evaluate many different feature subsets in order to identify the best subset. However, the computational complexity of training our non-linear classifiers forms a major bottleneck here, thus rendering such wrapper methods unfeasible in our current setup.

5. Conclusions

In this paper, we have demonstrated how machine learning polarity classifiers can benefit from novel features that capture structural aspects of natural language text. Typical machine learning approaches heavily rely on the presence of specific (groups of) words and as such inherently focus on *what* is said in a piece of text. However, as recent advances in rule-based sentiment analysis suggest that it may be more important *how* sentiment-carrying words are used in a text (as signaled by the text's rhetorical structure), we have proposed features that capture the sentiment of distinct rhetorical elements in a text, and we have evaluated the usefulness of these features on collections of English reviews in various domains.

Our experimental results over 10,000 English reviews suggest that the *what* and the *how* are both important cues for a text's polarity. Word-based features are indispensable to good polarity classifiers, yet structure-based sentiment information provides valuable additional guidance that can significantly improve the polarity classification performance of machine learning classifiers. In fact, the most informative features used by our best-performing classifiers capture the sentiment conveyed by specific rhetorical elements. Most of these elements constitute the core of a text, yet some elements provide crucial contextual information that does not constitute the core of a text.

Thus, we have successfully applied recent findings for rule-based sentiment analysis to a performance-wise more competitive machine learning approach to sentiment analysis. Our proposed richer vector representation of natural language text contributes to more effective automated sentiment analysis systems that can help better support decision making processes that require accurate insight into one's stakeholders' sentiment. Our findings, however, warrant several directions for future research.

A first direction for future research could be to validate our findings in other challenging sentiment analysis tasks like classifying polarity when figurative language – e.g., irony – is employed [61]. Second, other feature selection mechanisms and classifiers could be explored in order to further improve our performance. Last, the *what* and the *how* could be combined in future work, by differentiating word presence by rhetorical elements. For our current corpora, this is infeasible due to data sparsity issues that arise because of the high dimensionality of our data, compared to the number of instances in our corpora. Therefore, the usefulness of such features would need to be tested on a larger corpus, and with classifiers and feature selection mechanisms that can handle the substantially larger amount of data – with a much higher dimensionality – in a computationally efficient and effective way.

Acknowledgments

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

References

[1] Pingdom, Internet 2011 in Numbers, available online, `http://royal.pingdom.com/2012/01/17/internet-2011-in-numbers/` (2012).

[2] G. Mangnoesing, A. van Bunningen, A. Hogenboom, F. Hogenboom, F. Frasincar, An Empirical Study for Determining Relevant Features for Sentiment Summarization of Online Conversational Documents, in: 13th International Conference on Web Information Systems

Engineering (WISE 2012), Vol. 7651 of Lecture Notes in Computer Science, Springer, 2012, pp. 567–579.
[3] A. Hogenboom, B. Heerschop, F. Frasincar, U. Kaymak, F. de Jong, Multi-Lingual Support for Lexicon-Based Sentiment Analysis Guided by Semantics, Decision Support Systems 66 (1) (2014) 43–53.
[4] S. Madden, From Databases to Big Data, IEEE Internet Computing 16 (3) (2012) 4–6.
[5] S. Chan, Beyond Keyword and Cue-Phrase Matching: A Sentence-Based Abstraction Technique for Information Extraction, Decision Support Systems 42 (2) (2006) 759–777.
[6] C. Chang, C. Hsu, S. Lui, Automatic Information Extraction from Semi-Structured Web Pages by Pattern Discovery, Decision Support Systems 35 (1) (2003) 129–147.
[7] X. Wang, X. Jin, M. Chen, K. Zhang, D. Shen, Topic Mining over Asynchronous Text Sequences, IEEE Transactions on Knowledge and Data Engineering 24 (1) (2012) 156–169.
[8] A. Hogenboom, F. Hogenboom, F. Frasincar, K. Schouten, O. van der Meer, Semantics-Based Information Extraction for Detecting Economic Events, Multimedia Tools and Applications 64 (1) (2013) 27–52.
[9] A. Balahur, J. Hermida, A. Montoyo, Detecting Implicit Expressions of Emotion in Text: A Comparative Analysis, Decision Support Systems 53 (4) (2012) 742–753.
[10] R. Feldman, Techniques and Applications for Sentiment Analysis, Communications of the ACM 56 (4) (2013) 82–89.
[11] A. Montoyo, P. Martinez-Barco, A. Balahur, Subjectivity and Sentiment Analysis: An Overview of the Current State of the Area and Envisaged Developments, Decision Support Systems 53 (4) (2012) 675–679.
[12] A. Reyes, P. Rosso, Making Objective Decisions from Subjective Data: Detecting Irony in Customer Reviews, Decision Support Systems 53 (4) (2012) 754–760.
[13] B. Jansen, M. Zhang, K. Sobel, A. Chowdury, Twitter Power: Tweets as Electronic Word of Mouth, Journal of the American Society for Information Science and Technology 60 (11) (2009) 2169–2188.
[14] H. Rui, Y. Liu, A. Whinston, Whose and What Chatter Matters? The Effect of Tweets on Movie Sales, Decision Support Systems 55 (4) (2013) 863–870.
[15] X. Yu, Y. Liu, X. Huang, A. An, Mining Online Reviews for Predicting Sales Performance: A Case Study in the Movie Domain, IEEE Transactions on Knowledge and Data Engineering 24 (4) (2012) 720–734.
[16] Y. Yu, W. Duan, Q. Cao, The Impact of Social and Conventional Media on Firm Equity Value: A Sentiment Analysis Approach, Decision Support Systems 55 (4) (2013) 919–926.
[17] B. Pang, L. Lee, Opinion Mining and Sentiment Analysis, Foundations and Trends in Information Retrieval 2 (1) (2008) 1–135.
[18] B. Liu, Sentiment Analysis and Opinion Mining, Synthesis Lectures on Human Language Technologies, Morgan & Claypool Publishers, 2012.
[19] B. Heerschop, F. Goossen, A. Hogenboom, F. Frasincar, U. Kaymak, F. de Jong, Polarity Analysis of Texts using Discourse Structure, in: 20th ACM Conference on Information and Knowledge Management (CIKM 2011), Association for Computing Machinery, 2011, pp. 1061–1070.
[20] A. Hogenboom, M. Bal, F. Frasincar, D. Bal, U. Kaymak, F. de Jong, Lexicon-Based Sentiment Analysis by Mapping Conveyed Sentiment to Intended Sentiment, International Journal of Web Engineering and Technology 9 (2) (2014) 125–147.
[21] J. Chenlo, A. Hogenboom, D. Losada, Sentiment-Based Ranking of Blog Posts using Rhetorical Structure Theory, in: 18th International Conference on Applications of Natural Language to Information Systems (NLDB 2013), Vol. 7934 of Lecture Notes in Computer Science, Springer, 2013, pp. 13–24.
[22] A. Devitt, K. Ahmad, Sentiment Polarity Identification in Financial News: A Cohesion-based Approach, in: 45th Annual Meeting of the Association of Computational Linguistics (ACL 2007), Association for Computational Linguistics, 2007, pp. 984–991.
[23] M. Taboada, K. Voll, J. Brooke, Extracting Sentiment as a Function of Discourse Structure and Topicality, Tech. Rep. 20, Simon Fraser University, available online, `http://www.cs.sfu.ca/research/publications/techreports/#2008` (2008).
[24] M. Taboada, J. Brooke, M. Tofiloski, K. Voll, M. Stede, Lexicon-Based Methods for Sentiment Analysis, Computational Linguistics 37 (2) (2011) 267–307.
[25] V. Sauter, Decision Support Systems for Business Intelligence, 2nd Edition, Wiley, 2011.
[26] E. Cambria, B. Schuller, Y. Xia, C. Havasi, New Avenues in Opinion Mining and Sentiment Analysis, IEEE Intelligent Systems 28 (2) (2013) 15–21.
[27] J. Wiebe, T. Wilson, R. Bruce, M. Bell, M. Martin, Learning Subjective Language, Computational Linguistics 30 (3) (2004) 277–308.
[28] S. Baccianella, A. Esuli, F. Sebastiani, SentiWordNet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining, in: 7th Conference on International Language Resources and Evaluation (LREC 2010), European Language Resources Association, 2010, pp. 2200–2204.
[29] A. Hogenboom, D. Bal, F. Frasincar, M. Bal, F. de Jong, U. Kaymak, Exploiting Emoticons in Sentiment Analysis, in: 28th Symposium on Applied Computing (SAC 2013), ACM, 2013, pp. 703–710.
[30] A. Hogenboom, P. van Iterson, B. Heerschop, F. Frasincar, U. Kaymak, Determining Negation Scope and Strength in Sentiment Analysis, in: 2011 IEEE International Conference on Systems, Man, and Cybernetics (SMC 2011), IEEE, 2011, pp. 2589–2594.
[31] P. Chaovalit, L. Zhou, Movie Review Mining: A Comparison Between Supervised and Unsupervised Classification Approaches, in: 38th Hawaii International Conference on System Sciences (HICSS 2005), IEEE, 2005, p. 112c.
[32] B. Pang, L. Lee, S. Vaithyanathan, Thumbs up? Sentiment Classification using Machine Learning Techniques, in: 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP 2002), Association for Computational Linguistics, 2002, pp. 79–86.
[33] B. Pang, L. Lee, A Sentimental Education: Sentiment Analysis using Subjectivity Summarization based on Minimum Cuts, in: 42nd Annual Meeting of the Association for Computational Linguistics (ACL 2004), Association for Computational Linguistics, 2004, pp. 271–280.
[34] G. Paltoglou, M. Thelwall, A Study of Information Retrieval Weighting Schemes for Sentiment Analysis, in:

48th Annual Meeting of the Association for Computational Linguistics (ACL 2010), Association for Computational Linguistics, 2010, pp. 1386–1395.

[35] V. Hatzivassiloglou, J. Wiebe, Effects of Adjective Orientation and Gradability on Sentence Subjectivity, in: 18th International Conference on Computational Linguistics (COLING 2000), Association for Computational Linguistics, 2000, pp. 299–305.

[36] T. Mullen, N. Collier, Sentiment Analysis Using Support Vector Machines with Diverse Information Sources, in: 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP 2004), Association for Computational Linguistics, 2004, pp. 412–418.

[37] P. Turney, Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews, in: 40th Annual Meeting of the Association for Computational Linguistics (ACL 2002), Association for Computational Linguistics, 2002, pp. 417–424.

[38] C. Whitelaw, N. Garg, S. Argamon, Using Appraisal Groups for Sentiment Analysis, in: 14th ACM International Conference on Information and Knowledge Management (CIKM 2005), Association for Computing Machinery, 2005, pp. 625–631.

[39] D. Bal, M. Bal, A. van Bunningen, A. Hogenboom, F. Hogenboom, F. Frasincar, Sentiment Analysis with a Multilingual Pipeline, in: 12th International Conference on Web Information System Engineering (WISE 2011), Vol. 6997 of Lecture Notes in Computer Science, Springer, 2011, pp. 129–142.

[40] A. Hogenboom, M. Bal, F. Frasincar, D. Bal, Towards Cross-Language Sentiment Analysis through Universal Star Ratings, in: Seventh International Conference on Knowledge Management in Organizations (KMO 2012), Vol. 172 of Advances in Intelligent Systems and Computing, Springer, 2012, pp. 69–79.

[41] R. Mihalcea, C. Banea, J. Wiebe, Learning Multilingual Subjective Language via Cross-Lingual Projections, in: 45th Annual Meeting of the Association for Computational Linguistics (ACL 2007), Association for Computational Linguistics, 2007, pp. 976–983.

[42] Y. Wilks, M. Stevenson, The Grammar of Sense: Using Part-of-Speech Tags as a First Step in Semantic Disambiguation, Journal of Natural Language Engineering 4 (2) (1998) 135–143.

[43] A. Maas, R. Daly, P. Pham, D. Huang, A. Ng, C. Potts, Learning Word Vectors for Sentiment Analysis, in: 49th Annual Meeting of the Association for Computational Linguistics (ACL 2011), Association for Computational Linguistics, 2011, pp. 142–150.

[44] J. van der Meer, F. Boon, F. Hogenboom, F. Frasincar, U. Kaymak, A Framework for Automatic Annotation of Web Pages Using the Google Rich Snippets Vocabulary, in: 26th Symposium On Applied Computing (SAC 2011), Web Technologies Track, Association for Computing Machinery, 2011, pp. 765–772.

[45] A. Hogenboom, F. Boon, F. Frasincar, A Statistical Approach to Star Rating Classification of Sentiment, in: 1st International Symposium on Management Intelligent Systems (IS-MiS 2012), Vol. 171 of Advances in Intelligent Systems and Computing, Springer, 2012, pp. 251–260.

[46] L. Qu, G. Ifrim, G. Weikum, The Bag-of-Opinions Method for Review Rating Prediction from Sparse Text Patterns, in: 23rd International Conference on Computational Linguistics (COLING 2010), Association for Computational Linguistics, 2010, pp. 913–921.

[47] R. Socher, A. Perelygin, J. Wu, J. Chuang, C. Manning, A. Ng, C. Potts, Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank, in: 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP 2013), Association for Computational Linguistics, 2013, pp. 1631–1642.

[48] B. Chardon, F. Benamara, Y. Mathieu, V. Popescu, N. Asher, Measuring the Effect of Discourse Structure on Sentiment Analysis, in: 14th International Conference on Intelligent Text Processing and Computational Linguistics (CICLing 2013), Vol. 7817 of Lecture Notes in Computer Science, Springer, 2013, pp. 25–37.

[49] L. Polanyi, A. Zaenen, Computing Attitude and Affect in Text: Theory and Applications, Springer, 2006, Ch. Contextual Valence Shifters, pp. 1–10.

[50] C. Zirn, M. Niepert, H. Stuckenschmidt, M. Strube, Fine-Grained Sentiment Analysis with Structural Features, in: 5th International Joint Conference on Natural Language Processing (IJCNLP 2011), Asian Federation of Natural Language Processing, 2011, pp. 336–344.

[51] W. Mann, S. Thompson, Rhetorical Structure Theory: Toward a Functional Theory of Text Organization, Text 8 (3) (1988) 243–281.

[52] C. Fellbaum, WordNet: An Electronic Lexical Database, MIT Press, 1998.

[53] R. Soricut, D. Marcu, Sentence Level Discourse Parsing using Syntactic and Lexical Information, in: Human Language Technology and North American Association for Computational Linguistics Conference (HLT/NAACL 2003), Association for Computational Linguistics, 2003, pp. 149–156.

[54] J. Blitzer, M. Dredze, F. Pereira, Biographies, Bollywood, Boom-boxes and Blenders: Domain Adaptation for Sentiment Classification, in: 45th Annual Meeting of the Association of Computational Linguistics (ACL 2007), Association for Computational Linguistics, 2007, pp. 440–447.

[55] C. Manning, M. Surdeanu, J. Bauer, J. Finkel, S. Bethard, D. McClosky, The Stanford CoreNLP Natural Language Processing Toolkit, in: 52nd Annual Meeting of the Association for Computational Linguistics (ACL 2014), Association for Computational Linguistics, 2014, pp. 55–60.

[56] C. Manning, T. Grow, T. Grenager, J. Finkel, J. Bauer, Stanford Tokenizer, available online, `http://nlp.stanford.edu/software/tokenizer.shtml` (2010).

[57] J. Baldridge, T. Morton, OpenNLP, available online, `http://opennlp.sourceforge.net/` (2004).

[58] B. Walenz, J. Didion, OpenNLP, available online, `http://jwordnet.sourceforge.net/` (2008).

[59] I. Guyon, Mining Massive Data Sets for Security: Advances in Data Mining, Search, Social Networks and Text Mining, and their Applications to Security, IOS Press, 2008, Ch. Practical Feature Selection: From Correlation to Causality, pp. 27–43.

[60] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, I. Witten, The WEKA Data Mining Software: An Update, SIGKDD Explorations 11 (1) (2009) 10–18.

[61] A. Reyes, P. Rosso, On the Difficulty of Automatically Detecting Irony: Beyond a Simple Case of Negation, Knowledge and Information Systems 40 (3) (2014) 595–614.