Using Rhetorical Structure in Sentiment Analysis

Alexander Hogenboom[1]
hogenboom@ese.eur.nl

Flavius Frasincar[1]
frasincar@ese.eur.nl

Franciska de Jong[1,2]
f.m.g.dejong@utwente.nl

Uzay Kaymak[3]
u.kaymak@ieee.org

[1]Erasmus University Rotterdam, P.O. Box 1738, 3000 DR Rotterdam, the Netherlands
[2]Universiteit Twente, P.O. Box 217, 7500 AE Enschede, the Netherlands
[3]Eindhoven University of Technology, P.O. Box 513, 5600 MB Eindhoven, the Netherlands

ABSTRACT

Automated sentiment analysis has become an active field of study with a broad applicability. One of the key open research issues lies in dealing with structural aspects of text when analyzing its conveyed sentiment. Recent work uses structural aspects of text in order to distinguish important text segments from less important ones in terms of their contribution to the overall sentiment. Yet, existing methods are confined to making coarse-grained distinctions between text segments based on segments' rhetorical roles, while not accounting for the full hierarchical rhetorical structure in which these roles are defined. We hypothesize that a better understanding of a text's conveyed sentiment can be obtained by guiding automated sentiment analysis by the full rhetorical structure of text. We evaluate our hypothesis in a sentiment analysis framework based on Rhetorical Structure Theory, applied at the level of sentences, paragraphs, and documents. On an English movie review corpus, we obtain significant classification performance improvements compared to baselines not accounting for rhetorical structure, with the best results generated by exploiting a text's full sentential rhetorical structure.

Categories and Subject Descriptors

H.3.1 [**Information Storage and Retrieval**]: Content Analysis and Indexing—*Linguistic processing*; I.2.7 [**Artificial Intelligence**]: Natural Language Processing—*Language parsing and understanding*

General Terms

Algorithms, experimentation, performance

Keywords

Sentiment analysis, polarity classification, rhetorical structure, RST

1. INTRODUCTION

Popular Web sites like Twitter, Blogger, and Epinions allow their users to vent opinions on just about anything through an ever-increasing amount of short messages, blog posts, or reviews. Automated sentiment analysis techniques can extract traces of people's sentiment – i.e., people's attitude towards certain topics – from such texts [6]. This can yield competitive advantages for businesses [2], as one fifth of all tweets [10] and one third of all blog posts [15] discuss products or brands. Other potential applications for sentiment analysis tools lie in identifying the pros and cons of products [11] and analyzing the political domain [16].

Many commercial sentiment analysis systems mostly rely on the occurrence of sentiment-carrying words, but more sophistication can be introduced in several ways. Among the open research issues identified by Feldman [6] is the role of textual structure. Structural aspects may contain valuable information [19, 24] – sentiment-carrying words in a conclusion may contribute more to a text's overall sentiment than sentiment-carrying words in, e.g., background information.

Existing work typically uses structural aspects of text for distinguishing important text segments from less important ones in terms of their contribution to a text's overall sentiment, and subsequently weights a segment's conveyed sentiment in accordance with its importance. A segment's importance has often been related to its position in a text [19], yet recent methods make coarse-grained distinctions between segments based on their rhetorical roles [5, 8, 24] by applying Rhetorical Structure Theory (RST) [12].

While the potential of exploiting isolated rhetorical *roles* for sentiment analysis has been shown, the full rhetorical *structure* in which these roles are defined has thus far been ignored. Yet, a text segment with a rhetorical role can consist of smaller subordinate segments that are rhetorically related to one another, thus forming a hierarchical rhetorical tree structure. Important segments may contain less important parts. Since accounting for such structural aspects enables a better understanding of text [14], we hypothesize that guiding sentiment analysis by a deep analysis of a text's rhetorical structure can yield a better understanding of conveyed sentiment with respect to an entity of interest.

Our contribution is three-fold. First, as an alternative to existing, shallow RST-guided sentiment analysis approaches that typically focus on rhetorical relations in top-level splits of RST trees, we propose to focus on the leaf nodes of RST trees or, alternatively, to account for the full RST trees.

Second, we propose a novel RST-based weighting scheme, which is more refined than existing weighting schemes [8, 24]. Third, rhetorical relations across sentences and paragraphs can provide useful context for the rhetorical structure of their subordinate text segments. Therefore, whereas existing work mostly guides sentiment analysis by sentence-level analyses of rhetorical structure (if at all), we additionally incorporate paragraph-level and document-level analyses of rhetorical structure into the process. We thus account for rhetorical relations across sentences and paragraphs.

The remainder of this paper is structured as follows. In Section 2, we review how rhetorical relations are typically exploited in automated sentiment analysis. Then, we propose and evaluate our novel approach to sentiment analysis guided by a deep analysis of rhetorical structure in Sections 3 and 4, respectively. We conclude in Section 5.

2. SENTIMENT ANALYSIS AND RHETORICAL RELATIONS

Automated sentiment analysis is related to natural language processing, computational linguistics, and text mining. Typical tasks include distinguishing subjective text segments from objective ones and determining the polarity of words, sentences, text segments, or documents [18]. We address the binary document-level polarity classification task, dealing with classifying documents as positive or negative.

The state-of-the-art in sentiment analysis has been reviewed extensively [6, 18]. Existing methods range from machine learning methods, exploiting patterns in vector representations of text, to lexicon-based methods, accounting for individual words' semantic orientation by matching these words with a sentiment lexicon, listing words and their associated sentiment. Lexicon-based methods are typically more robust across domains and texts [24], and allow for intuitively incorporating deep linguistic analyses.

Deep linguistic analysis is a key success factor for sentiment analysis systems, as it helps dealing with compositionality [21], i.e., the way in which the semantic orientation of text is determined by the combined semantic orientations of its constituent phrases. This compositionality can be captured by accounting for the grammatical [20] or discursive [4, 5, 8, 22, 24] structure of text.

RST [12] is a popular discourse analysis framework. It splits text into rhetorically related segments that may in turn be split too, thus yielding a hierarchical rhetorical structure. Each segment is classified as either a nucleus or a satellite. Nuclei form a text's core, supported by satellites that are considered less important. In total, 23 types of relations exist between RST elements [12]. For example, a satellite may form an elaboration on or a contrast with a nucleus.

Consider the positive sentence "*While always complaining that he hates this type of movies, John bitterly confessed that he enjoyed this movie.*", containing mostly negative words. RST can split this sentence into a hierarchical rhetorical structure of text segments, shown in Figure 1. The top-level nucleus contains the core message ("*John bitterly confessed that he enjoyed this movie.*"), with a satellite providing background information. This background satellite consists of a nucleus ("*he hates this type of movies,*") and an attributing satellite ("*While always complaining that*"). Similarly, the core consists of a nucleus ("*he enjoyed this movie.*") and an attributing satellite ("*John bitterly confessed that*"). The sentence conveys a positive overall sentiment towards the movie, due to the way in which the sentiment-carrying words are used in the sentence – the actual sentiment is conveyed by the nucleus "*he enjoyed this movie.*".

Rhetorical relations have already been used in sentiment analysis, with some methods [5, 8, 24] relying more strongly on relations defined in RST than others [4, 22]. In order to identify rhetorical relations in text, the most successful works use the Sentence-level PArsing of DiscoursE (SPADE) tool [23], which creates an RST tree for each sentence. Another parser is the HIgh-Level Discourse Analyzer [9] (HILDA), which parses discourse structure at document level by means of a greedy bottom-up tree-building method that uses machine-learning classifiers to iteratively assign RST relation labels to those (compound) segments of a document that are most likely to be rhetorically related. SPADE and HILDA take as input free text – a priori divided into paragraphs and sentences – and produce LISP-like representations of the text and its rhetorical structure.

Document-level polarity classification has been successfully guided by analyses of the most relevant text segments, as identified by differentiating between top-level nuclei and satellites in sentence-level RST trees [24]. Another, more elaborate method of utilizing RST in sentiment analysis accounts for the different types of relations between nuclei and satellites [8], which yields improvements over distinguishing nuclei from satellites only [5, 8].

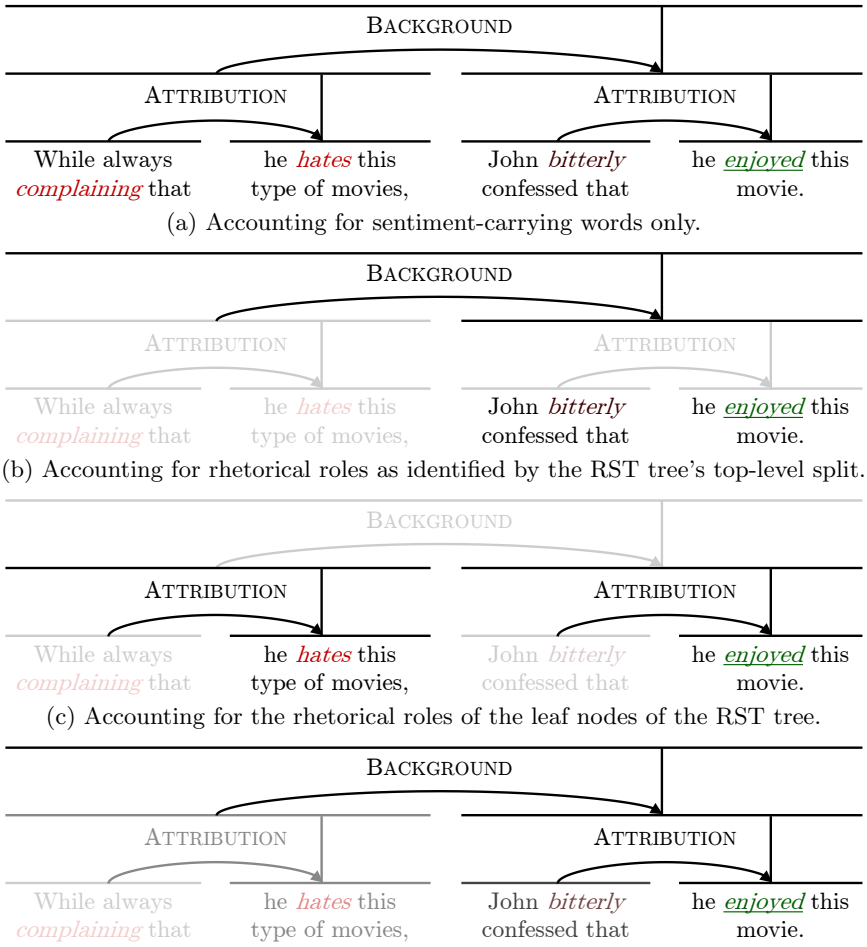
3. POLARITY CLASSIFICATION GUIDED BY RHETORICAL STRUCTURE

Existing methods do not use RST to its full extent, yet typically focus on top-level splits of sentence-level RST trees and thus employ a rather shallow, coarse-grained analysis. However, rhetorical relations are defined within a hierarchical structure, modeled by RST trees. Nuclei may, e.g., contain less important satellites, which should be treated as such. Therefore, we propose to guide polarity classification by a deep analysis of a text's hierarchical rhetorical *structure* rather than its isolated rhetorical *relations*. We account for rhetorical relations within and across sentences by allowing not only for sentence-level, but also paragraph-level and document-level analyses of rhetorical structure.

3.1 Fine-Grained Analysis

Figure 1 illustrates the potential of RST-guided polarity classification. Based on the sentiment-carrying words alone, our example sentence can best be classified as negative (see Figure 1(a)). Accounting for the rhetorical roles of text segments as identified by the RST tree's top-level split enables a more elaborate, but still coarse-grained analysis of the overall sentiment (see Figure 1(b)). The top-level nucleus contains as many positive as negative words and may hence be classified as either positive or negative. The negative words in the top-level satellite trigger a negative classification of this segment, which is a potentially irrelevant segment that should be assigned a lower weight in the analysis.

However, such a coarse-grained analysis does not capture the nuances of lower-level splits of an RST tree. For instance, the top-level nucleus of our example consists of two segments, one of which is the sentence's actual core ("*he enjoyed this movie.*"), whereas the other is less relevant and should therefore be assigned a lower weight in the analysis.



(a) Accounting for sentiment-carrying words only.

(b) Accounting for rhetorical roles as identified by the RST tree's top-level split.

(c) Accounting for the rhetorical roles of the leaf nodes of the RST tree.

(d) Accounting for the full, hierarchical rhetorical structure.

Figure 1: Interpretations of a positive RST-structured sentence, consisting of nuclei – marked with vertical lines – and satellites. Negative words are printed red, in italics, whereas positive words are underlined and printed green, in italics. Sentiment-carrying words with a relatively high intensity are brighter. Horizontal lines signal the spans of the RST elements on each level of the hierarchical rhetorical structure. Arrows and their (capitalized) captions represent the relations of satellite elements to nucleus elements. Text segments and RST elements that are assigned a relatively low weight in the analysis of the conveyed sentiment are more transparent than those that receive higher weights.

Accounting for the rhetorical roles of the leaf nodes of an RST tree rather than the top-level splits can thus enable a more accurate sentiment analysis (see Figure 1(c)).

Yet, a focus on leaf nodes of RST trees alone does not account for the text segments' rhetorical roles being defined within the context of the rhetorical roles of the segments that embed them. For instance, the second leaf node in our example RST tree is a nucleus of a possibly irrelevant background satellite, whereas the fourth leaf node is the nucleus of the nucleus and forms the actual core. The full rhetorical structure should be considered in the analysis in order to account for this (see Figure 1(d)).

We propose a lexicon-based sentiment analysis framework that can perform an analysis of the rhetorical structure of a piece of text at various levels of granularity and that can subsequently use this information for classifying the text's overall polarity. Our framework, visualized in Figure 2, takes several steps in order to classify the polarity of a document.

3.2 Word-Level Sentiment Scoring

Our method first splits a document into paragraphs, sentences, and words. Then, for each sentence, the Part-of-Speech (POS) and lemma of each word is determined. The word sense of each word is subsequently disambiguated using an unsupervised algorithm that iteratively selects the word sense with the highest semantic similarity to the word's context [8]. The sentiment of each word, associated with its particular combination of POS, lemma, and word sense is then retrieved from a sentiment lexicon like SentiWordNet [1].

3.3 Rhetorical Structure Processing

In order to guide the polarity classification of a document d by its rhetorical structure, sentiment scores are computed for each segment s_i. Our framework supports several methods of computing such scores, i.e., a baseline method plus several RST-based methods that can be applied to sentence-level, paragraph-level, and document-level RST trees.

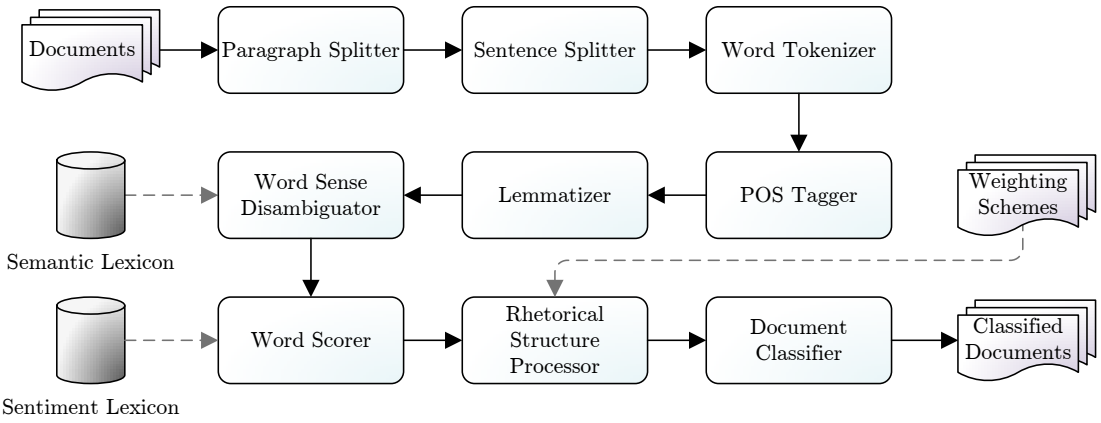


Figure 2: A schematic overview of our sentiment analysis framework. Solid arrows signal the information flow, whereas dashed arrows indicate a used-by relationship.

3.3.1 Baseline

As a baseline, we consider text segments to be the sentences S_d of document d, with their associated baseline sentiment score $\zeta^B_{s_i}$ being the weighted sum of the score ζ_{t_j} of each word t_j and its weight w_{t_j}, i.e.,

$$\zeta^B_{s_i} = \sum_{t_j \in s_i} \left(\zeta_{t_j} \cdot w_{t_j}\right), \quad \forall s_i \in S_d. \tag{1}$$

3.3.2 Top-Level Rhetorical Structure

Our framework additionally supports the top-level RST-based method applied in existing work. We refer to this approach as T. The sentiment score $\zeta^T_{s_i}$ of a top-level RST segment s_i is defined as the sum of the sentiment ζ_{t_j} associated with each word t_j in segment s_i, weighted with a weight w_{s_i} associated with the segment's rhetorical role, i.e.,

$$\zeta^T_{s_i} = \sum_{t_j \in s_i} \left(\zeta_{t_j} \cdot w_{s_i}\right), \quad \forall s_i \in T_d, \tag{2}$$

with T_d representing all top-level RST nodes in the RST trees for document d.

3.3.3 Leaf-Level Rhetorical Structure

Another method is our leaf-level RST-based analysis L. The sentiment score $\zeta^L_{s_i}$ of an RST segment s_i from the leaf nodes L_d of an RST tree for document d is computed as the summed sentiment score of its words, weighted for the segment's rhetorical role, i.e.,

$$\zeta^L_{s_i} = \sum_{t_j \in s_i} \left(\zeta_{t_j} \cdot w_{s_i}\right), \quad \forall s_i \in L_d. \tag{3}$$

3.3.4 Hierarchical Rhetorical Structure

The last supported approach is our method of accounting for the full path from an RST tree root to a leaf node, such that the sentiment conveyed by the latter can be weighted while accounting for its rhetorical context. In our hierarchy-based sentiment scoring method H, we model the sentiment score $\zeta^H_{s_i}$ of a leaf-level RST segment s_i as a function of the sentiment scores of its words and the weights w_{r_n} associated with the rhetorical role of each node r_n from the nodes P_{s_i} on the path from the root to the leaf, i.e.,

$$\zeta^H_{s_i} = \sum_{t_j \in s_i} \zeta_{t_j} \cdot \left(\frac{\sum_{r_n \in P_{s_i}} \left(|w_{r_n}| \cdot \delta^{-\left(\lambda_{r_n}-1\right)} \right)}{\sum_{r_n \in P_{s_i}} \delta^{-\left(\lambda_{r_n}-1\right)}} \right) \cdot \prod_{r_n \in P_{s_i}} \operatorname{sgn}\left(w_{r_n}\right), \quad \forall s_i \in L_d, \quad \delta > 1, \tag{4}$$

where δ represents a diminishing factor and λ_{r_n} signals the level of node r_n in the RST tree, with the level of the root node being 1. For $\delta > 1$, each subsequent level contributes less than its parent does to the segment's RST-based weight, thus preserving the hierarchy of the relations in the path.

3.4 Classifying Document Polarity

The segment-level sentiment scores can be aggregated in order to determine the overall polarity of a document. The baseline, top-level, leaf-level, and hierarchy-based sentiment scores ζ^B_d, ζ^T_d, ζ^L_d, and ζ^H_d for document d are defined as

$$\zeta^B_d = \sum_{s_i \in S_d} \zeta^B_{s_i}, \tag{5}$$

$$\zeta^T_d = \sum_{s_i \in T_d} \zeta^T_{s_i}, \tag{6}$$

$$\zeta^L_d = \sum_{s_i \in L_d} \zeta^L_{s_i}, \tag{7}$$

$$\zeta^H_d = \sum_{s_i \in L_d} \zeta^H_{s_i}. \tag{8}$$

The resulting document-level sentiment scores can be used to classify document d's polarity c_d as negative (-1) or positive (1), following

$$c_d = \begin{cases} -1 & \text{if } (\zeta_d - \epsilon) < 0, \\ 1 & \text{else.} \end{cases} \tag{9}$$

Here, ϵ represents an offset that corrects a possible bias in the sentiment scores caused by people's tendency to write negative reviews with rather positive words [24], i.e.,

$$\epsilon = 0.5 \left(\frac{\sum_{d \in \Phi} \zeta_d}{|\Phi|} + \frac{\sum_{d \in N} \zeta_d}{|N|} \right), \tag{10}$$

with Φ and N denoting the respective subsets of positive and negative documents in a training set.

3.5 Weighting Schemes

We consider six different weighting schemes. Two serve as baselines and are applicable to the baseline sentiment scoring approach as defined in (1) and (5). The other schemes apply to our RST-based approaches as defined in (2) and (6), in (3) and (7), and in (4) and (8).

Our BASELINE scheme serves as an absolute baseline and assigns each word a weight of 1 – structural aspects are not accounted for. A second baseline is a position-based scheme. In this POSITION scheme, word weights are uniformly distributed and range from 0 for the first word to 1 for the last word of a text, as an author's views are more likely to be summarized near the end of a text [19].

The first RST-based scheme (I) assigns a weight of 1 to nuclei and a weight of 0 to satellites [24]. The second RST-based scheme (II) matches the second set of weights for nuclei and satellites used by Taboada et al. [24], i.e., 1.5 and 0.5, respectively. In both schemes I and II, we set the diminishing factor δ for the H method to 2, such that each level in a tree is at least as important as all of its subsequent levels combined, thus enforcing a strict hierarchy.

Another RST-specific scheme is the extended scheme X, in which we differentiate satellite weights by their RST relation type [8]. Additionally, we propose a novel extension of X, i.e., the full weighting scheme F, in which we not only differentiate satellite weights, but also nucleus weights by their RST relation type. For both X and F, the weights and the diminishing factor δ can be optimized.

4. EVALUATION

The variants of our polarity classification approach guided by structural aspects of text are evaluated by means of a set of experiments. We focus on a widely used collection of 1,000 positive and 1,000 negative English movie reviews [17].

4.1 Experimental Setup

In the Java implementation of our framework, we detect paragraphs using the `<P>` and `</P>` tags in the HTML files. For sentence splitting, we rely on the preprocessed reviews [17]. The Stanford Tokenizer [13] is used for word segmentation. For POS tagging and lemmatization, we use the OpenNLP [3] POS tagger and the JWNL API [25], respectively. We link word senses to WordNet [7], thus enabling the retrieval of their sentiment scores from SentiWordNet [1] by subtracting the associated negativity scores from the positivity scores. This yields real numbers ranging from -1 (negative) to 1 (positive). In the final aggregation of word scores, each word is assigned a weight by means of one of our methods (detailed in Table 1), most of which are RST-based.

We evaluate our methods on the accuracy, precision, recall, and F_1-score on positive and negative documents, and the overall accuracy and macro-level F_1-score. We assess the statistical significance of performance differences by means of paired, one-sided t-tests, comparing methods against one another in terms of their mean performance measures over all ten folds, under the null hypothesis that the mean performance of a method is less than or equal to the mean performance of another method.

We apply ten-fold cross-validation. For each fold, we optimize the offsets, the weights, and the diminishing factor δ for weighting schemes X and F using particle swarm optimization [5], with particles searching a solution space, where their coordinates correspond with the weights and offsets (between -2 and 2), and the diminishing factor δ (between 1 and 2). The fitness of a particle is its macro-level F_1-score on a training set.

Table 1: Characteristics of our considered polarity classification approaches, i.e., our baselines, our SPADE-based sentence-level RST-guided methods, and our HILDA-based sentence-level, paragraph-level, and document-level RST-guided methods.

Method	Unit	RST	Weights
BASELINE	Document	None	BASELINE
POSITION	Document	None	POSITION
SPADE.S.T.I	Sentence	Top-level	I
SPADE.S.T.II	Sentence	Top-level	II
SPADE.S.T.X	Sentence	Top-level	X
SPADE.S.T.F	Sentence	Top-level	F
SPADE.S.L.I	Sentence	Leaf-level	I
SPADE.S.L.II	Sentence	Leaf-level	II
SPADE.S.L.X	Sentence	Leaf-level	X
SPADE.S.L.F	Sentence	Leaf-level	F
SPADE.S.H.I	Sentence	Hierarchical	I
SPADE.S.H.II	Sentence	Hierarchical	II
SPADE.S.H.X	Sentence	Hierarchical	X
SPADE.S.H.F	Sentence	Hierarchical	F
HILDA.S.T.I	Sentence	Top-level	I
HILDA.S.T.II	Sentence	Top-level	II
HILDA.S.T.X	Sentence	Top-level	X
HILDA.S.T.F	Sentence	Top-level	F
HILDA.S.L.I	Sentence	Leaf-level	I
HILDA.S.L.II	Sentence	Leaf-level	II
HILDA.S.L.X	Sentence	Leaf-level	X
HILDA.S.L.F	Sentence	Leaf-level	F
HILDA.S.H.I	Sentence	Hierarchical	I
HILDA.S.H.II	Sentence	Hierarchical	II
HILDA.S.H.X	Sentence	Hierarchical	X
HILDA.S.H.F	Sentence	Hierarchical	F
HILDA.P.T.I	Paragraph	Top-level	I
HILDA.P.T.II	Paragraph	Top-level	II
HILDA.P.T.X	Paragraph	Top-level	X
HILDA.P.T.F	Paragraph	Top-level	F
HILDA.P.L.I	Paragraph	Leaf-level	I
HILDA.P.L.II	Paragraph	Leaf-level	II
HILDA.P.L.X	Paragraph	Leaf-level	X
HILDA.P.L.F	Paragraph	Leaf-level	F
HILDA.P.H.I	Paragraph	Hierarchical	I
HILDA.P.H.II	Paragraph	Hierarchical	II
HILDA.P.H.X	Paragraph	Hierarchical	X
HILDA.P.H.F	Paragraph	Hierarchical	F
HILDA.D.T.I	Document	Top-level	I
HILDA.D.T.II	Document	Top-level	II
HILDA.D.T.X	Document	Top-level	X
HILDA.D.T.F	Document	Top-level	F
HILDA.D.L.I	Document	Leaf-level	I
HILDA.D.L.II	Document	Leaf-level	II
HILDA.D.L.X	Document	Leaf-level	X
HILDA.D.L.F	Document	Leaf-level	F
HILDA.D.H.I	Document	Hierarchical	I
HILDA.D.H.II	Document	Hierarchical	II
HILDA.D.H.X	Document	Hierarchical	X
HILDA.D.H.F	Document	Hierarchical	F

Table 2: Performance measures of our considered baselines, our SPADE-based sentence-level RST-guided methods, and our HILDA-based sentence-level, paragraph-level, and document-level RST-guided methods, based on 10-fold cross-validation on the movie review data set. The best performance in each group of methods is printed in bold for each performance measure.

	Positive			Negative			Overall	
Method	Precision	Recall	F_1	Precision	Recall	F_1	Accuracy	F_1
BASELINE	0.632	0.689	0.659	0.658	0.599	0.627	0.644	0.643
POSITION	0.637	0.713	0.673	0.674	0.593	0.631	0.653	0.652
SPADE.S.T.I	0.638	0.675	0.656	0.655	0.617	0.635	0.646	0.646
SPADE.S.T.II	0.640	0.688	0.663	0.663	0.613	0.637	0.651	0.650
SPADE.S.T.X	**0.693**	**0.725**	**0.709**	**0.712**	**0.679**	**0.695**	**0.702**	**0.702**
SPADE.S.T.F	0.703	0.726	0.715	0.717	0.694	0.705	0.710	0.710
SPADE.S.L.I	0.636	0.702	0.667	0.667	0.598	0.631	0.650	0.649
SPADE.S.L.II	0.640	0.700	0.669	0.669	0.607	0.637	0.654	0.653
SPADE.S.L.X	0.699	0.715	0.707	0.708	0.692	0.700	0.704	0.703
SPADE.S.L.F	0.705	0.731	0.718	0.721	0.694	0.707	0.713	0.712
SPADE.S.H.I	0.647	0.678	0.662	0.662	0.630	0.645	0.654	0.654
SPADE.S.H.II	0.642	0.696	0.668	0.668	0.612	0.639	0.654	0.653
SPADE.S.H.X	0.707	0.723	0.715	0.716	**0.700**	0.708	0.712	0.711
SPADE.S.H.F	**0.710**	**0.738**	**0.724**	**0.727**	0.699	**0.713**	**0.719**	**0.718**
HILDA.S.T.I	0.633	0.676	0.654	0.652	0.608	0.629	0.642	0.642
HILDA.S.T.II	0.636	0.686	0.660	0.659	0.607	0.632	0.647	0.646
HILDA.S.T.X	0.692	0.709	0.701	0.702	0.685	0.693	0.697	0.697
HILDA.S.T.F	0.697	**0.745**	**0.720**	**0.726**	0.676	0.700	0.711	0.710
HILDA.S.L.I	0.629	0.685	0.656	0.654	0.596	0.624	0.641	0.640
HILDA.S.L.II	0.636	0.685	0.660	0.659	0.608	0.632	0.647	0.646
HILDA.S.L.X	0.698	0.711	0.705	0.706	0.693	0.699	0.702	0.702
HILDA.S.L.F	**0.705**	0.732	0.718	0.721	0.693	**0.707**	**0.713**	**0.712**
HILDA.S.H.I	0.634	0.675	0.654	0.653	0.611	0.631	0.643	0.643
HILDA.S.H.II	0.638	0.688	0.662	0.661	0.609	0.634	0.649	0.648
HILDA.S.H.X	0.699	0.693	0.696	0.695	**0.701**	0.698	0.697	0.697
HILDA.S.H.F	0.699	0.740	0.719	0.724	0.682	0.702	0.711	0.711
HILDA.P.T.I	0.618	0.638	0.628	0.626	0.605	0.615	0.622	0.621
HILDA.P.T.II	0.628	0.674	0.650	0.648	0.600	0.623	0.637	0.637
HILDA.P.T.X	0.681	0.697	0.689	0.690	0.674	0.682	0.686	0.685
HILDA.P.T.F	0.703	0.702	0.702	0.702	0.703	0.703	0.703	0.702
HILDA.P.L.I	0.632	0.684	0.657	0.656	0.602	0.628	0.643	0.642
HILDA.P.L.II	0.633	0.685	0.658	0.657	0.603	0.629	0.644	0.643
HILDA.P.L.X	0.690	0.705	0.697	0.698	0.683	0.691	0.694	0.694
HILDA.P.L.F	0.701	**0.720**	**0.710**	**0.712**	0.693	0.702	0.707	0.706
HILDA.P.H.I	0.583	0.609	0.596	0.591	0.565	0.578	0.587	0.587
HILDA.P.H.II	0.629	0.682	0.655	0.653	0.598	0.624	0.640	0.639
HILDA.P.H.X	0.706	0.683	0.694	0.693	0.716	0.704	0.700	0.699
HILDA.P.H.F	**0.713**	0.692	0.703	0.701	**0.722**	**0.711**	**0.707**	**0.707**
HILDA.D.T.I	0.627	0.616	0.621	0.622	0.633	0.628	0.625	0.624
HILDA.D.T.II	0.627	0.650	0.639	0.637	0.614	0.625	0.632	0.632
HILDA.D.T.X	0.682	0.689	0.685	0.686	0.678	0.682	0.684	0.683
HILDA.D.T.F	0.684	0.696	0.690	0.691	0.679	0.685	0.688	0.687
HILDA.D.L.I	0.627	0.679	0.652	0.650	0.596	0.622	0.638	0.637
HILDA.D.L.II	0.631	0.687	0.658	0.656	0.598	0.626	0.643	0.642
HILDA.D.L.X	0.689	0.719	0.704	0.706	0.675	0.690	0.697	0.697
HILDA.D.L.F	0.701	**0.727**	**0.714**	**0.717**	0.690	0.703	**0.709**	**0.708**
HILDA.D.H.I	0.580	0.516	0.546	0.564	0.627	0.594	0.572	0.570
HILDA.D.H.II	0.630	0.663	0.646	0.645	0.611	0.627	0.637	0.637
HILDA.D.H.X	0.706	0.696	0.701	0.700	**0.710**	0.705	0.703	0.703
HILDA.D.H.F	**0.707**	0.708	0.707	0.707	0.706	**0.707**	0.707	0.707

4.2 Experimental Results

Our experimental analysis consists of four steps. First, we compare the performance of our considered methods in Section 4.2.1. We then analyze the optimized weights and diminishing factors in Section 4.2.2 and we demonstrate how documents are typically perceived by distinct methods in Section 4.2.3. Last, we discuss some caveats for our findings in Section 4.2.4.

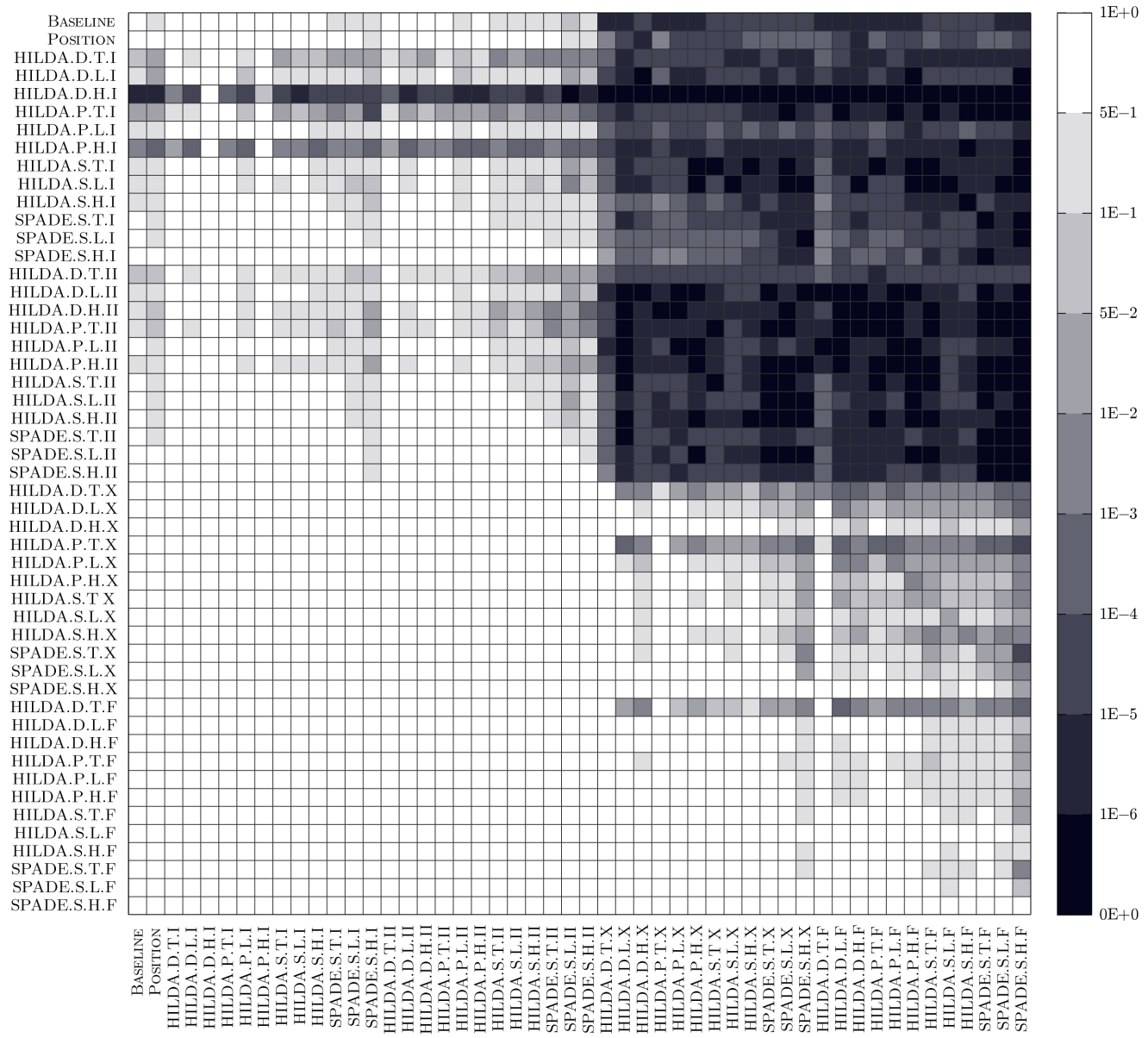


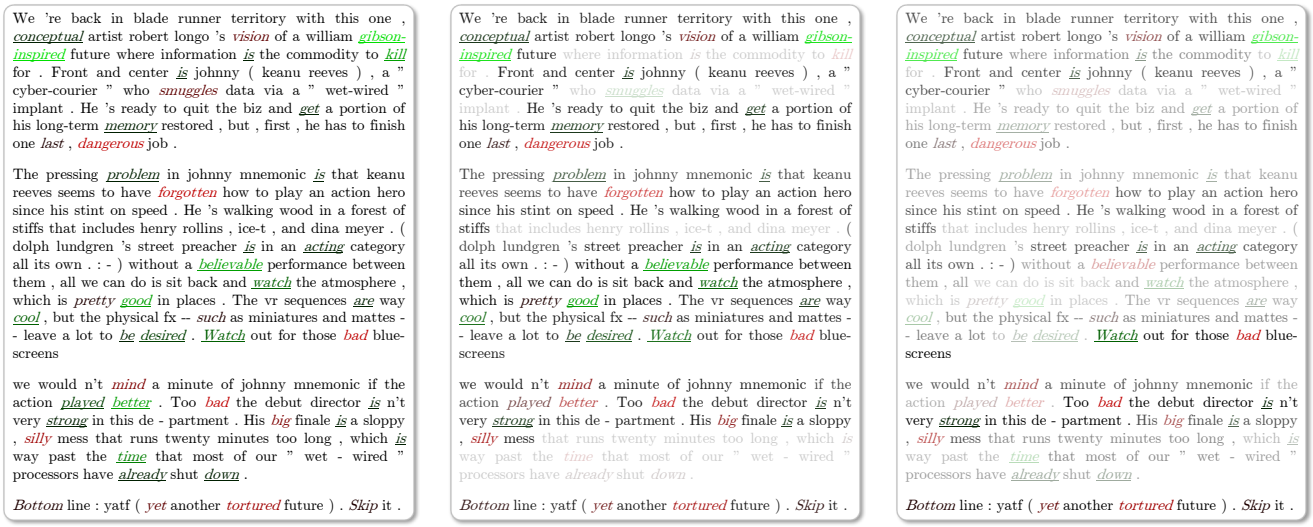
Figure 3: The p-values for the paired, one-sided t-test assessing the null hypothesis of the mean macro-level F_1-scores of the methods in the columns being smaller than or equal to the mean macro-level F_1-scores of the methods in the rows.

4.2.1 Performance

Table 2 presents our methods' performance, whereas Figure 3 visualizes the p-values for the macro-level F_1-scores for all combinations of methods, ordered from top to bottom and from left to right in accordance with an initial ranking from the worst to the best performing methods. We have sorted the methods based on their weighting scheme (Baseline, Position, I, II, X, and F), their analysis level (HILDA.D, HILDA.P, HILDA.S, and SPADE.S), and their RST analysis method (T, L, and H), respectively. Darker colors indicate lower p-values, with darker rows and columns signaling weak and competitive approaches, respectively. Given a correct ordering, darker colors should be right from the diagonal, towards the upper right corner of Figure 3.

Three trends can be observed in Table 2 and Figure 3. First, weighting schemes X and F significantly outperform the I, II, Baseline, and Position schemes, with F significantly outperforming X in most cases. Conversely, schemes I, II, Baseline, and Position do not exhibit clearly significant performance differences with respect to one another.

A second trend is that methods guided by document-level RST trees are typically outperformed by comparable methods utilizing paragraph-level RST trees. Sentence-level RST trees yield the best performance. An explanation lies in misclassifications of rhetorical relations being potentially more harmful in larger RST trees – a misclassified relation in one of the top levels of a document-level RST tree can cause a misinterpretation of a large part of the document.



(a) Baseline. (b) SPADE.S.T.X. (c) SPADE.S.H.F.

Figure 4: Movie review cv817_3675, as processed by various sentiment analysis methods. Negative words are printed red, in italics, whereas positive words are underlined and printed green, in italics. Sentiment-carrying words with high intensity are brighter, whereas words carrying less sentiment are darker. Text segments that are assigned a relatively low weight in the analysis of the conveyed sentiment are more transparent than text segments with higher weights.

The third trend is that methods applying the hierarchy-based RST analysis method H typically slightly outperform comparable approaches that use the leaf-based analysis L instead, which in turn significantly outperform comparable approaches that use the top-level RST analysis method T. Clearly, the deeper analyses L and H yield a significant advantage over the rather shallow analysis method T.

Some methods stand out in particular. First, HILDA.D.T and HILDA.P.T are relatively weak, especially when using weighting schemes I and II. The top-level RST analysis method T is typically too coarse-grained for larger RST trees, as, e.g., documents are segmented in only two parts when using document-level RST trees. Other weak methods are HILDA.D.H.I and HILDA.P.H.I. The combination of the naive weighting scheme I with a deep, hierarchy-based analysis of the rhetorical structure of a document or its paragraphs results in a narrow focus on very specific segments.

Approaches that stand out positively are those applying the hierarchy-based RST analysis H to sentence-level RST trees, with weighting schemes X and F, i.e., HILDA.S.H.X, HILDA.S.H.F, SPADE.S.H.X, and SPADE.S.H.F. These approaches perform comparably well because they involve a detailed analysis of rhetorical structure – the analysis is performed on the smallest considered units (sentences), the hierarchy of RST trees is accounted for, and the weights are differentiated per type of rhetorical relation. This confirms our hypothesis that the sentiment conveyed by a text can be captured more adequately by incorporating a deep analysis of the text's rhetorical structure.

4.2.2 *Optimized Weights*

The optimized weights for distinct types of nuclei and satellites, as defined in RST [12], exhibit several patterns. Nucleus elements are generally rather important in weighting scheme X, with most weights ranging between 0.5 and 1. The sentiment expressed in elaborating satellites is typically assigned a similar importance. Contrasting satellites mostly receive weights around or below 0. Background satellites are typically assigned relatively low weights as well.

In weighting scheme F, nuclei and satellites in an attributing relation are typically both assigned weights around 0.5. Conversely, for background and contrasting relations, satellites are more clearly distinct from nuclei. Background satellites are typically assigned less importance than their associated nuclei, with respective weights around 0 and 1. For contrasting relations, nuclei are mostly assigned weights between 0.5 and 1, whereas optimized satellite weights are typically negative.

The optimized values for the diminishing factor δ are typically around 1.5 and 1.25 for weighting schemes X and F, respectively. This results in the first 15 (for X) or 30 (for F) levels of RST trees being accounted for. Interestingly, with some (document-level) RST trees being over 100 levels deep in our corpus, the optimized diminishing factors effectively mostly disregard the lower, most fine-grained parts of RST trees, thus realizing a balance between the level of detail and potential noise in the analysis.

4.2.3 *Processing a Document*

The observed differences in performance of our polarity classification methods originate in how these methods perceive a document in the sentiment analysis process. Figure 4 demonstrates how the interpretations of a movie review differ across various methods.

Our software assigns many words in our example a positive sentiment, whereas fewer words are identified as carrying negative sentiment. When assigning each part of the review an equal weight (see Figure 4(a)), this relative abundance of positive words suggests that the review is rather positive, whereas it is negative.

Our best performing RST-based baseline, SPADE.S.T.X, considers only the top-level splits of sentence-level RST trees and yields a rather coarse-grained segmentation of the sentences, thus leaving little room for nuances (see Figure 4(b)). Nevertheless, SPADE.S.T.X successfully highlights the conclusion and the arguments supporting this conclusion. The polarity of some positive words is even inverted, as the reviewer uses them in a negative way.

SPADE.S.H.F, our best performing approach, yields a very detailed analysis in which subtle distinctions are made between small text segments (see Figure 4(c)). These nuances help bringing out the parts of the review that are most relevant with respect to its overall sentiment. SPADE.S.H.F ignores most of the irrelevant background information in the first paragraph and highlights the reviewer's main concerns in the second and third paragraphs. Moreover, sentiment related to the movie's good aspects is often inverted and mostly ignored. The overall recommendation is emphasized in the last paragraph. All in all, it is the incorporation of the detailed analysis of the review's rhetorical structure into the sentiment analysis process that facilitates a better understanding of the review by our polarity classifier.

4.2.4 Caveats

A failure analysis reveals that some misclassifications of authors' sentiment are caused by sarcasm and proverbs occasionally being misinterpreted. Additionally, not all sentiment-carrying words are identified by SentiWordNet or correctly disambiguated. Other challenges are related to the information content of documents. For instance, some reviewers tend to mix their opinionated statements with plot details that contain irrelevant sentiment-carrying words. Additionally, some reviewers evaluate a movie by comparing it with other movies. Such statements require a distinction between entities and their associated sentiment, as well as real-world knowledge to be incorporated into the analysis.

Even though our RST-based polarity classification methods cannot cope particularly well with the specific phenomena mentioned above, they significantly outperform our non-RST baselines. However, these improvements come at a cost of processing time being increased with about a factor 10. The bottleneck here is formed by the RST parsers, rather than the application of our weighting schemes.

5. CONCLUSIONS

We have demonstrated that sentiment analysis can benefit from a deep analysis of a text's rhetorical *structure*, enabling the distinction between important text segments and less important ones in terms of their contribution to a text's overall sentiment. This is a significant step forward with respect to existing work, which is limited to guiding sentiment analysis by shallow analyses of rhetorical *relations* in (mostly sentence-level) rhetorical structure trees.

Our contribution is three-fold. First, our novel polarity classification methods guided by deep leaf-level or hierarchy-based RST analyses significantly outperform existing approaches that are guided by shallow RST analyses, or by no RST-based analyses at all. Second, the novel RST-based weighting scheme in which we differentiate the weights of nuclei and satellites by their RST relation type significantly outperforms existing schemes. Third, we have compared the performance of polarity classification approaches guided by sentence-level, paragraph-level, and document-level RST trees, thus revealing that RST-based polarity classification works best when focusing on RST trees of smaller units of a text, such as sentences.

In future work, we aim to investigate the applicability of other, possibly more scalable methods for exploiting (discourse) structure of text in sentiment analysis. Additionally, we plan to validate our findings on other corpora, covering other domains or other types of text.

6. ACKNOWLEDGMENTS

Special thanks go to Bas Heerschop and Frank Goossen for their contributions in the early stages of this work. The authors of this paper are partially supported by the Dutch national program COMMIT.

7. REFERENCES

[1] S. Baccianella, A. Esuli, and 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)*, pages 2200–2204. European Language Resources Association, 2010.

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

[3] J. Baldridge and T. Morton. OpenNLP, 2004. Available online, `http://opennlp.sourceforge.net/`.

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

[5] J. Chenlo, A. Hogenboom, and D. Losada. Rhetorical Structure Theory for Polarity Estimation: An Experimental Study. *Data and Knowledge Engineering*, 2014. Available online, `http://dx.doi.org/10.1016/j.datak.2014.07.009`.

[6] R. Feldman. Techniques and Applications for Sentiment Analysis. *Communications of the ACM*, 56(4):82–89, 2013.

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

[8] B. Heerschop, F. Goossen, A. Hogenboom, F. Frasincar, U. Kaymak, and F. de Jong. Polarity Analysis of Texts using Discourse Structure. In *20th ACM Conference on Information and Knowledge Management (CIKM 2011)*, pages 1061–1070. Association for Computing Machinery, 2011.

[9] H. Hernault, H. Prendinger, D. duVerle, and M. Ishizuka. HILDA: A Discourse Parser Using Support Vector Machine Classification. *Dialogue and Discourse*, 1(3):1–33, 2010.

[10] B. Jansen, M. Zhang, K. Sobel, and A. Chowdury. Twitter Power: Tweets as Electronic Word of Mouth. *Journal of the American Society for Information Science and Technology*, 60(11):2169–2188, 2009.

[11] S. Kim and E. Hovy. Automatic Identification of Pro and Con Reasons in Online Reviews. In *21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics (COLING/ACL 2006)*, pages 483–490. AAAI, 2006.

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

[13] C. Manning, T. Grow, T. Grenager, J. Finkel, and J. Bauer. Stanford Tokenizer, 2010. Available online, `http://nlp.stanford.edu/software/tokenizer.shtml`.

[14] D. Marcu. The Rhetorical Parsing of Unrestricted Texts: A Surface-Based Approach. *Computational Linguistics*, 26(3):395–448, 2000.

[15] P. Melville, V. Sindhwani, and R. Lawrence. Social Media Analytics: Channeling the Power of the Blogosphere for Marketing Insight. In *1st Workshop on Information in Networks (WIN 2009)*, 2009. Available online, `http://www.prem-melville.com/publications/`.

[16] T. Mullen and R. Malouf. A Preliminary Investigation into Sentiment Analysis of Informal Political Discourse. In *2006 AAAI Symposium on Computational Approaches to Analyzing Weblogs (AAAI-CAAW 2006)*, pages 159–162. AAAI, 2006.

[17] B. Pang and 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)*, pages 271–280. Association for Computational Linguistics, 2004.

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

[19] B. Pang, L. Lee, and S. Vaithyanathan. Thumbs up? Sentiment Classification using Machine Learning Techniques. In *2002 Conference on Empirical Methods in Natural Language Processing (EMNLP 2002)*, pages 79–86. Association for Computational Linguistics, 2002.

[20] A. Sayeed, J. Boyd-Graber, B. Rusk, and A. Weinberg. Grammatical Structures for Word-Level Sentiment Detection. In *2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL HLT 2012)*, pages 667–676. Association for Computational Linguistics, 2012.

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

[22] S. Somasundaran, G. Namata, J. Wiebe, and L. Getoor. Supervised and Unsupervised Methods in Employing Discourse Relations for Improving Opinion Polarity Classification. In *2009 Conference on Empirical Methods in Natural Language Processing (EMNLP 2009)*, pages 170–179. Association for Computational Linguistics, 2009.

[23] R. Soricut and 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)*, pages 149–156. Association for Computational Linguistics, 2003.

[24] M. Taboada, K. Voll, and J. Brooke. Extracting Sentiment as a Function of Discourse Structure and Topicality. Technical Report 20, Simon Fraser University, 2008. Available online, `http://www.cs.sfu.ca/research/publications/techreports/#2008`.

[25] B. Walenz and J. Didion. OpenNLP, 2008. Available online, `http://jwordnet.sourceforge.net/`.