Representing and Resolving Negation for Sentiment Analysis

Emanuele Lapponi Jonathon Read Lilja Øvrelid
Department of Informatics
University of Oslo
Email: {emanuel,jread,liljao}@ifi.uio.no

***Abstract*—Proper treatment of negation is an important characteristic of methods for sentiment analysis. However, while there is a growing body of research on the automatic resolution of negation, it is not yet clear as to how negation is best represented for different applications. To begin to address this issue, we review representation alternatives and present a state-of-the-art system for negation resolution that is interoperable across these schemes. By employing different configurations of this system as a component in a testbed for lexically-based sentiment classification, we demonstrate that the choice of representation can have a significant impact on downstream processing.**

I. Introduction

Negation is a strongly expressive linguistic phenomenon. In its most obvious instance, it reverses the truth value of a statement (Examples 1 and 2), while in more subtle examples it is a conveyor of euphemisms and irony [1].

(1) I did enjoy the movie.
(2) I did not enjoy the movie.

Recent years have witnessed a growing interest in the treatment of negation within the field of Natural Language Processing. The fact that a sentence is negated and furthermore which parts of a sentence the negation pertains to constitute relevant information for several applications, including information extraction, textual entailment and sentiment analysis.

In this work we focus on the application of a system for negation resolution to polarity classification as a case study. However, the technique we present is also beneficial for the broader tasks of opinion mining and affect recognition. For example, negation resolution can complement common-sense based approaches (e.g. [2]) by appropriately inverting analyses conducted using ontologies and reasoning tools.

While much of the early work on the representation and resolution of negation has been focused on the biomedical domain [3]–[5], an increasing number of negation-annotated corpora taken from other, quite different domains have recently been made available to the research community. Research in this field has been further spurred by research competitions, so-called shared tasks, dedicated to this topic. Most recently the *SEM 2012 shared task dealt with the task of negation resolution [6]. The representation of negation and, in particular, the scope of negation, varies quite notably across different corpora and there is little consensus in the field as to how negation should best be annotated. It is unclear as of yet whether these differences in annotation practice have practical consequence for performance down-stream.

We present a system for negation resolution what was among the top performers at the recent *SEM 2012 shared task [7]. We augment the system for interoperability across various schemes of negation annotation. The two annotation schemes contrasted in this paper represent quite different approaches to the scope of negation, and have been applied to different textual domains. This enables a study of the effects of negation annotation and domain differences for sentiment analysis.

We begin with a review of representations of negation and some techniques for their automatic application in Section II. In Section III we present our own approach to negation resolution, with results of experiments comparing it to the current state-of-the-art. Section IV shows the effects of using the different schemes in the context of a simple testbed for lexicon-based sentiment classification, and Section V presents our conclusions and plans for future work.

II. Representing and Resolving Negation

Recent work in analyzing negation has been prompted by the availability of the BioScope (BS) collection of biomedical text annotated with negation [3]. Annotations in this corpus indicate two aspects of negations: *cues* (operators of negation, e.g. no, not, never) and *scopes* (the span of a sentence actually negated). Scopes in the BS corpus are *maximal*, meaning that they encompass the largest syntactic unit possible, and furthermore always include the cue. In the running example below (where angle brackets ($\langle\rangle$) denote the negation cue, while curly brackets ($\{\}$) indicate scopes) the scope of *not* in (3) is the entire sentence.

(3) $\{$I did $\langle$not$\rangle$ enjoy the movie$\}$. [BS]

The Product Reviews (PR) data set [8] differs from BioScope in regard to both domain and annotation scheme. Developed with the aim of representing negation in web text, it is comprised of reviews obtained from Google Product Search. Being composed of product reviews, this dataset is of particular interest to the sentiment analysis community. We note that negation occurs in 32.2% of sentences in PR, indicating the importance of negation resolution to sentiment analysis.

In this dataset negation cues are not annotated explicitly, while scope annotations are *minimal* and are intended to cover the semantic portion of the text that is negated. Below, PR-style annotation is applied to our running example:

(4) I did not {enjoy the movie}. [PR]

Most recently, a dataset was released in conjunction with the *SEM 2012 shared task on negation resolution [6]. With data taken from the novels of Conan Doyle (CD), the dataset is similar to BS as the maximal scope is annotated, but is distinct in several aspects: cues are not part of the scope; morphological cues are annotated (e.g. ⟨im⟩possible); and scopes can be discontinuous. Furthermore, the concept of scope is complemented by a notion of *negated events*. These are the events/states inside scopes which are semantically negated. For instance, in (5) CD annotation is applied to our running example (where the negated event is underlined).

(5) {I did} ⟨not⟩ {enjoy the movie}. [CD]

Note though, that the event annotation is only present in sentences that are factual—events in instances of imperatives, conditionals, suppositions etc. are not annotated.

Negation resolution is typically broken down into subtasks. It is first necessary to identify instances of negation cues. This is typically achieved using a static lexicon of known cues [8], or with a supervised machine learning classifier [4], [5], [9], often complemented by observations of the ambiguity of cues in training data.

The second subtask is that of scope resolution. A variety of techniques have been employed to resolve the scope, such as: predicting whether tokens are at the beginning or end of a negation scope using a meta-learning setup [4]; applying conditional random fields to estimate the most probable sequence of labels (i.e. negated or not) for a given sentence [8]; heuristics operating over dependency analyses [5] or constituent trees [10]; and an automatically-learned discriminative ranking function over syntactic constituents [5], [10].

When attempting to reproduce the annotations in the Conan Doyle corpus a third task is necessary—namely, the identification of negated events. The best-performing system in the event subtask of the *SEM shared task first filtered non-factual negations using a bag-of-words based support vector machine factuality classifier before using a discriminative ranker to select among in-scope words, with features of paths in constituent trees [10].

III. Sequence Labeling the Scope of Negation using Dependency Features

This section presents our system for negation resolution, which is based on a previous submission to the *SEM shared task [7], but is augmented for interoperability across schemes of negation annotation described in Section II. The heart of the system is the application of conditional random field models for sequence labeling which makes use of a rich set of lexical and syntactic features, together with a fine-grained set of labels that capture the scopal behavior of tokens.

The system accepts sentences annotated with negation cues (which can be provided by any external method) as input. First, the raw text is processed and enriched with lexical (tokens, parts-of-speech, and lemmas from the Natural Language Toolkit [11]) and syntactic information (dependency analyses from MaltParser [12]). Next, the data is converted for input into a package for sequence labeling (Wapiti [13]). Finally, the output of the labeler is post-processed to transform token labels to the appropriate scheme.

The following subsections describe aspects of this system in more detail before evaluating the system with some experiments on negation resolution that utilize the CD and PR datasets. In the following, we focus on the annotations provided in these two corpora for two main reasons: first, they exemplify markedly different annotation strategies with respect to negation, and second, these two datasets are taken from domains that resemble that of our use case more than biomedical text, however, the difference in domain is true across these two corpora as well; PR consists of user-generated content from the web, while CD contains early twentieth century English novellas.

A. Format Conversion

Previous work on sequence labeling for negation resolution has typically applied two labels to tokens in a sentence—inside or outside a negation scope [8]. However, as different kinds of label transitions may affect the decision of the sequence labeler, we attempt to capture the behavior of specific tokens within the mechanics of negation not only with feature modeling, but also with a finer-grained label set. In this section we discuss this label set, and how CD and PR datasets are converted into a common format for processing.

The token-wise annotations in CD contain multiple layers of information. Tokens may or may not be negation cues and they can be either in or out of scope; in-scope tokens may or may not be negated events, and are associated with each of the cues they are negated by. Moreover, scopes may be (partially) overlapping, as in Fig. 1, where the scope of *without* is contained within the scope of *never*. We convert this representation internally by assigning one of six labels to each token: O, CUE, MCUE, N, E and S, for out-of-scope, cue, morphological (affixal) cue, in-scope, event and negation stop respectively. The rationale behind the separation of cues in two classes is that morphological cues show scopal properties that are quite different from other negation cues. Fig. 1 shows the result of the conversion to sequence labels for a sentence from the CD corpus.

The conversion into label sequences is more straightforward for the PR dataset, which does not contain negated events or overlapping scopes. The representation is essentially the same as for CD, with the difference that there is no MCUE label, since there are no morphological cues in the corpus. As cues are not provided in the gold data, these are instead automatically labeled, where the first token to the left of annotated scopes returns a match from a lexicon of negation cues. In the few scope-containing sentences where the lexicon lookup is unsuccessful, gold cues are manually annotated.

B. Training a Sequence Labeler

Using the information from the gold annotations and preprocessing we record several features to train conditional random

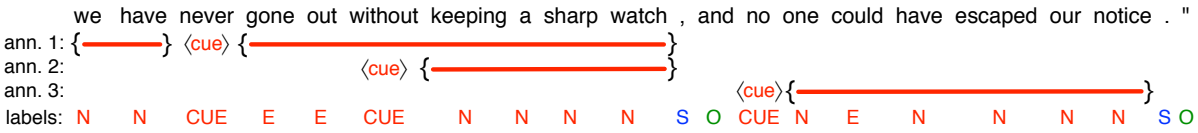

Fig. 1. A sentence from CD showing the annotation-to-label conversion.

	Features
Lexical	Token Lemma Part-of-speech Forward token bigram and trigram Backward token bigram and trigram Forward part-of-speech trigram Backward part-of-speech trigram Lexicalized part-of-speech Forward Lexicalized part-of-speech bigram Backward Lexicalized part-of-speech bigram
Syntactic	Constituent Dependency relation First order head part-of-speech Second order head part-of-speech Lexicalized dependency relation Part-of-Speech disambiguated dependency relation
Cue-dependent	Left Token distance Right Token distance Directed dependency distance Bidirectional dependency distance Dependency path Lexicalized dependency path

TABLE I
FEATURES USED TO TRAIN THE CONDITIONAL RANDOM FIELD MODELS

field models.[1] These features (summarized in Table I) may be divided into lexical, syntactic and cue-dependent subsets. The features employed extend on those used by previous work on negation resolution with conditional random fields [8] in several ways. For example, rather than recording surface bigrams, trigrams and parts-of-speech in only one direction, we look in both directions. Furthermore, parts-of-speech and dependency relations are both combined and lexicalized.

Figure 2 provides a simplified example of the feature extraction, showing the extraction of a backward part-of-speech trigram, the right token distance, and dependency path feature.

The features extracted via the dependency graphs aim at modeling the syntactic relationship between each token and the closest negation cue. Parsing each sentence with MaltParser returns dependency graphs—non-linear syntactic representations that can be explored to find more general traits that characterize in-scope tokens. The direction of the edges in these graphs is based on the notion of syntactic *heads*, from which edges originate, and their *dependents* [14].

With the token indices as unambiguous identifiers and

[1]After some experimentation with joint labeling of scopes and events, we opted for separation of the two models, and hence train separate models for the two tasks of scope resolution and event detection. In the model for scopes, all event labels are switched to in-scope; conversely, in-scope tokens become out-of-scope tokens in the event model.

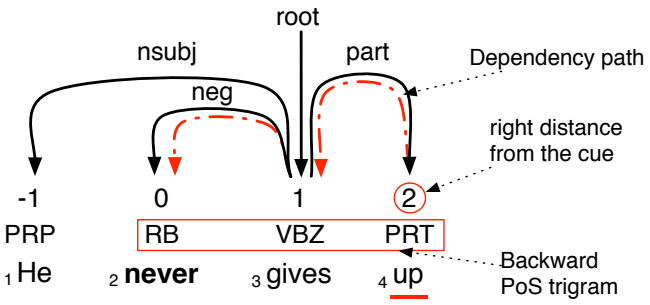

Fig. 2. In this example sentence showing the active *right token distance* and *backward token trigram* features when the currently processed token is *up*. Looking at the accompanying dependency graph, we see that the active *dependency path* feature is $\uparrow\ part \downarrow\ neg$.

the head relations being incoming edges, we represent the dependency graph as a set V of vertices and two different sets of edges, E and E′—the former containing only the directed edges and the latter containing also the reversed. For Figure 2, we have:

$$\begin{aligned} \mathrm{V} &= \{1, 2, 3, 4\} \\ \mathrm{E} &= \{\langle 3, 1\rangle, \langle 3, 2\rangle, \langle 3, 4\rangle\} \\ \mathrm{E}' &= \{\langle 3, 1\rangle, \langle 3, 2\rangle, \langle 3, 4\rangle, \langle 1, 3\rangle, \langle 2, 3\rangle, \langle 4, 3\rangle\} \end{aligned}$$

From the graph G = {V, E} we extract the shortest path from the *head* of the negation cue to all other vertices in the graph; we start from the head of a cue because, being directed acyclic graphs essentially trees, negation cues are very often found in the leaves, hence would yield no path from the cue to anywhere in the tree.

From G′ = {V, E′} we extract the shortest path from the negation to all other vertices. If there is more than one negation cue in the sentence, the relevant cue is the lexically closest one to a token with no intermediate punctuation. To compute the shortest paths we employ an unweighted implementation of the Dijkstra algorithm [15].

We furthermore use these shortest paths to record the *Dependency Graph Path* as a feature. This feature was inspired by the Parse Tree Path feature presented in the context of semantic role labeling [16]. It represents the path traversed from each token to the cue, encoding both the dependency relations and the direction of the arc that is traversed: for instance, the syntactic relation between *our* and *no* in Figure 1 is described as $\uparrow\ poss \uparrow\ dobj \downarrow\ nsubj \downarrow\ det$.

Finally, following previous research [8], we also encode the part-of-speech of the first and second order syntactic head of

	Configuration		PCS
(A)	Councill *et al.* reported [8]	classified cues	39.80
(B)	Councill *et al.* reproduced	classified cues	42.37
(C)	Our system	classified cues	48.53
(D)	Our system	gold cues	67.85

TABLE II
COMPARATIVE RESULTS USING THE PR DATASET

System	CNS
UiO_1 [10]	43.83
Our system	41.28
FBK [18]	35.74
UWashington [19]	34.04
UMichigan [20]	27.23

TABLE III
COMPARATIVE RESULTS ON THE CD DATASET

System		P	R	F_1
UiO_1	Scopes	83.89	60.64	70.39
	Events	60.58	75.00	67.02
Our system	Scopes	85.71	62.65	72.39
	Events	66.90	57.40	61.79

TABLE IV
*SEM SHARED TASK SCOPE AND EVENT RESOLUTION RESULTS

each token. For the token *no* in Figure 1, for instance, we record the PoS of *one* and *escaped*, respectively.

C. Experiments

We now present the results of our negation resolution system and compare our results to other state-of-the-art systems utilizing the PR and CD datasets. For discussion of experiments assessing the impact of the features, see previous work [17].

1) Product Reviews: Following previous research with the PR dataset, we measure performance of our system using the *Percentage of Correct Scopes* (PCS), which is calculated as the "number of correct spans divided by the number of true spans" [8]. Our evaluation was cross-validated with ten folds. Results for the PCS score reported by previous research are shown in Table II Row (A), together with our attempt at replicating their setup in Row (B). The differences observed are due to different pre-processing components and evaluation scripts. In this respect the results in Row (B) are a special configuration of our system that uses only the O and N labels and does not take advantage of features other than the ones reported by Councill *et al.* for the CRF model. Both (B) and (C) systems use a lexicon-lookup to identify cues of negation. We note that the extended label and feature set of our system offers a significant[2] improvement ($p < 0.05$). However, a shortcoming of this approach is the cue classification component, as evidenced by (D), which shows the performance of the system when using gold cues.

2) Conan Doyle: To evaluate the system's performance on the CD dataset we follow the methodology established by the *SEM 2012 shared task, which established subsets of data for training, development and final evaluation. To measure performance we employ the evaluation script provided by the shared task, which measures precision, recall and F_1 over cues, scopes and events. Additionally, the script measures overall performance with *Correct Negation Sentences* (CNS), which looks at the proportion of sentences containing negations where cues, scopes and negated events are perfectly resolved.

In these experiments, negation cues were identified using a support vector machine classifier trained using features of n-grams over word forms and lemmas. The classifier assumes that the set of negation cues is a closed class. This has the advantage of greatly reducing complexity of the learning task, which in turn results in improved performance. The approach achieved an F_1 of 91.31 when detecting cues in CD [10].

Table III lists the top performing systems in the closed portion of the *SEM 2012 shared task. UiO_1, the top performing system, aligns scopes to constituents, using a machine-learning model to choose a syntactic unit among different candidates [10]. This system addresses discontinuous scopes directly during post-processing with rules applied on the constituents. Additionally, UiO_1 has a dedicated factuality classifier, which is used to discard event-labeling in non-factual contexts. FBK [18] and UWashington [19] and UMichigan [20] both use a CRF classifier to resolve scopes and events, though we observe that our more involved approach in terms of feature and label selection does make a rather marked difference, with over 4 percentage points better CNS than FBK.

Table IV displays scope and event resolution results for our two systems in comparison with UiO_1 (note, we can not reliably compare scope and event performance with other systems, as they are affected by different approaches to cue classification). We find that our system outperforms the UiO_1 system by two percentage points in the scope resolution task. Comparing UiO_1 event resolution results to our system, we see the positive effects of their aforementioned factuality classifier. Our system has higher precision, which means that it is better at not generating false positives, but is far worse in terms of recall, being nearly 23 percentage points behind UiO_1.

While this is a shortcoming with respect to the *SEM shared task, its importance for sentiment analysis is less clear. Recall that in the CD dataset negated events are only annotated in factual contexts—meaning that instances of modals, suppositions, future tense etc. are not annotated. We speculate that this is not desirable for sentiment analysis, in which these phenomena play a vital role.

IV. A CASE STUDY OF NEGATION FOR SENTIMENT ANALYSIS

Sentiment analysis is an active field of research that focuses on the automatic detection and treatment of opinion in natural language processing applications. It is important for many reasons, including: recommendation systems, market research,

[2]Significance was estimated using the Sign test over instances of negation.

detection of criminal communication, political opinion measurement and affective computing.

Recently attention in the community has turned to the resolution of negation to assist in sentiment analysis tasks. In its most straightforward instance, the application of negation to sentiment analysis tasks lies in polarity reversal; the polarity of the statement *I did enjoy the movie* should be the opposite of its negation *I did not enjoy the movie*. This issue has been tackled with a variety of approaches, for example by detecting cues and mapping all features between the cue and the next punctuation token to negated features (i.e. *I did not enjoy the movie* becomes *I did not NOT_enjoy NOT_the NOT_movie*. However, this simple approach does not have a significant impact on the performance of machine learning methods for sentiment classification [21]. There is however, a consensus across many approaches that negation scope is important [22], and indeed its benefit has been demonstrated empirically [8].

As discussed in Section II, the notion of negation scope is different in the two corpora that have been used in this work—maximal scopes in CD and minimal scopes in PR. How do these two different takes on scope affect the application of negation resolution to tasks in sentiment analysis? This section presents a testbed system designed to address this question, and uses it to extrinsically evaluate the performance of different representations of negation.

A. A Testbed for Sentiment Analysis

Sentiment classification is performed using a scoring function that takes advantage of the AFINN-111 lexicon of 2,477 manually annotated sentiment-bearing words [23]. Each entry in this lexicon is annotated with an integer indicating the degree of sentiment it conveys, with -5 being most negative and +5 being most positive. We sum the lexicon's score (s) of each word (w) in a text (T). However, if a word is found to be inside the scope of a negation we inverse the score. We also consider the count of positive and negative matches, $c\,(\text{pos})$ and $c\,(\text{neg})$:

$$\text{score}\,(T) = c\,(\text{pos}) - c\,(\text{neg}) + \sum_{w \in T} \begin{cases} \text{is negated}\,(w) & -s\,(w) \\ \text{else} & +s\,(w) \end{cases}$$

A text is then deemed positive if the score is greater than zero and negative if it is less than zero.

While such a simple approach is far from the start-of-the-art in sentiment classification, it does allow us to simply and effectively evaluate how our negation resolution system performs as a polarity inverter, and how its training data and representation of negation affects sentiment classification.

B. Experiment

To evaluate our system we employ the Polarity v.2.0 dataset, a collection of 1,000 positive and 1,000 negative movie reviews obtained from the Internet Movie Database [24].

Various configurations of the system described in Section III were evaluated with polarity-wise F_1 score. We consider two baseline configurations: (1) not applying any negation resolution, and (2) applying a simple approach that assumes negation scopes from the cue to the next punctuation token. We compare these to one configuration that utilizes negation scopes obtained from our PR-based model for polarity reversal, while two more are based on the CD-model and use scopes and negated event respectively. False positives are counted when a review is classified with the wrong polarity; rather than assigning reviews where the score from the lexicon sums to zero to one of the two classes, we count such outcomes as false negatives. To ensure comparability of results across schemes, for all configurations we identify cues by consulting a lexicon.

	positive	**negative**
total matches	43582	40527
PR-based scope inversion	3065	1000
CD-based scopes inversion	4274	4822
CD-based events inversion	789	897

TABLE V
FREQUENCIES OF AFINN-111 MATCHES IN POLARITY V2.0

Configuration		**P**	**R**	**F_1**	cmb **F_1**
Baseline 1	pos	79.65	99.12	88.33	75.95
	neg	47.11	97.69	63.57	
Baseline 2	pos	77.65	98.58	86.87	77.90
	neg	53.34	97.40	68.93	
PR-scopes	pos	78.00	98.84	87.19	78.62
	neg	54.77	97.11	70.04	
CD-scopes	pos	77.38	98.19	86.55	79.29
	neg	57.04	97.74	72.04	
CD-events	pos	79.25	99.11	88.08	77.32
	neg	50.35	98.22	66.57	

TABLE VI
RESULTS OF THE DOCUMENT-LEVEL SENTIMENT CLASSIFIER

Table V shows the total amount of matches from the lexicon for both positive and negative reviews and how many times their score is inverted in each negation-aware configuration. In this respect, they all perform as expected, with CD-*scopes* being the most inversion-eager configuration, CD-*events* the most conservative and PR-*scopes* intermediate.

Table VI shows results for all configurations. At a glance, we see that the system is biased towards positive polarity, with F_1 score for positive reviews being consistently higher than it is for the negative ones. All configurations that incorporate polarity inversion outperform the baseline for negative reviews, while none outperforms the positive baseline. CD-scopes obtains the best overall performance, improving on the negative baseline by almost 10 percentage points. We also note that the configurations using scopes from our approach to negation resolution outperform the simple-negation baseline, indicating that simplistic approaches to negation are insufficient for sentiment classification.

These results are particularly noteworthy because the PR-

based system is trained on text that is much more similar to that found in the reviews we are classifying, which intuitively should make it perform better in this domain. In fact, although the PR corpus is in part conceived with sentiment analysis in mind, the maximal, syntactically motivated scopes from the CD corpus work significantly[3] better for polarity inversion in the context of negative documents ($p < 0.05$).

The same CD configuration, however, is also the weakest on positive reviews, at almost 2 percentage points lower F_1 than the baseline; the converse is true for the CD-events configuration. Here, the F_1 gain for negative reviews is 3 percentage points, while the loss for positive reviews is less than 0.2 points. The performance of the PR-scopes configuration lies somewhere in between the other two in all respects.

V. Conclusion

In this paper we have reviewed different schemes for representing negation and presented a state-of-the art system for negation resolution that is interoperable across schemes. Using this system as a component in a simple negation-aware testbed for sentiment classification enabled us to assess the impact of different schemes of negation annotation. We found that the choice of representation can have a significant effect on the performance of the sentiment classifier.

Although all negation-aware configurations are beneficial in terms of the combined F_1 score, it is interesting that no configuration yields any improvement on positive review classification, while the benefits for negative reviews seem to be proportional to the breadth of the negation scopes generated by the different models. This could be related to the way negation is used in positive and negative contexts, or simply to the fact that our naive approach does not harness the benefits of negation scope resolution fully. In future work we intend to examine these questions with finer-grained experiments (e.g. [25]). For instance, how should nested negations be handled by the sentiment system?

We will also consider the application of the BS scheme of annotation. While on the surface this is similar to the annotation found in the CD dataset, the guidelines of the respective datasets suggest they could be quite different; annotations in BS are syntactic in nature, while in CD they are to a larger extent semantically motivated.

Acknowledgment

Our thanks go to Ryan McDonald, Roser Morante, Finn Årup Nielsen and Bo Pang for sharing their data. Thanks also to our colleagues at UiO and the anonymous reviewers for their comments.

References

[1] L. R. Horn, *The Expression of Negation*. Walter de Gruyter, 2010.

[2] E. Cambria and A. Hussain, *Sentic Computing: Techniques, Tools and Applications*. Dordrecht, Netherlands: Springer, 2012.

[3] V. Vincze, G. Szarvas, R. Farkas, G. Móra, and J. Csirik, "The BioScope corpus: Biomedical texts annotated for uncertainty, negation and their scopes," *BMC Bioinformatics*, vol. 9(Suppl 11), 2008.

[3] Significance was estimated using the Sign test over documents.

[4] R. Morante and W. Daelemans, "A metalearning approach to processing the scope of negation," in *Proceedings of the 13th Conference on Natural Language Learning*, Boulder, CO, USA, June 2009, p. 21–29.

[5] E. Velldal, L. Øvrelid, J. Read, and S. Oepen, "Speculation and negation: Rules, rankers and the role of syntax," *Computational Linguistics*, vol. 38, no. 2, p. 369–410, 2012.

[6] R. Morante and E. Blanco, "*SEM 2012 shared task: Resolving the scope and focus of negation," in *Proceedings of the 1st Joint Conference on Lexical and Computational Semantics*, Montréal, Canada, June 2012, p. 265–274.

[7] E. Lapponi, E. Velldal, L. Øvrelid, and J. Read, "UiO2: Sequence-labeling negation using dependency features," in *Proceedings of the 1st Joint Conference on Lexical and Computational Semantics*, Montréal, Canada, June 2012, p. 319–327.

[8] I. Councill, R. McDonald, and L. Velikovich, "What's great and what's not: Learning to classify the scope of negation for improved sentiment analysis," in *Proceedings of the Workshop on Negation and Speculation in Natural Language Processing*, Uppsala, Sweden, July 2010, p. 51–59.

[9] Q. Zhu, J. Li, H. Wang, and G. Zhou, "A unified framework for scope learning via simplified shallow semantic parsing," in *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, Cambridge, MA, USA, October 2010, p. 714–724.

[10] J. Read, E. Velldal, L. Øvrelid, and S. Oepen, "UiO1: Constituent-based discriminative ranking for negation resolution," in *Proceedings of the 1st Joint Conference on Lexical and Computational Semantics*, Montréal, Canada, June 2012, p. 310–318.

[11] S. Bird, E. Klein, and E. Loper, *Natural Language Processing with Python*. Beijing: O'Reilly, 2009.

[12] J. Nivre, J. Hall, J. Nilsson, A. Chanev, G. Eryigit, S. Kübler, S. Marinov, and E. Marsi, "MaltParser: A language-independent system for data-driven dependency parsing," *Natural Language Engineering*, vol. 13, no. 2, 2007.

[13] T. Lavergne, O. Cappé, and F. Yvon, "Practical very large scale CRFs," in *Proceedings of the 48th Meeting of the Association for Computational Linguistics*, Uppsala, Sweden, July 2010, p. 504–513.

[14] A. M. Zwicky, "Heads," *Journal of Linguistics*, vol. 21, pp. 1–29, 1985.

[15] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische Mathematik*, vol. 1, p. 269–271, 1959.

[16] D. Gildea and D. Jurafsky, "Automatic labeling of semantic roles," *Computational Linguistics*, vol. 28, p. 245–288, 2002.

[17] E. Lapponi, "Why not! sequence labeling the scope of negation using dependency features," Master's thesis, University of Oslo, August 2012.

[18] M. F. M. Chowdhury, "FBK: Exploiting phrasal and contextual clues for negation scope detection," in *Proceedings of the 1st Joint Conference on Lexical and Computational Semantics*, Montréal, Canada, June 2012, p. 340–346.

[19] J. P. White, "UWashington: Negation resolution using machine learning methods," in *Proceedings of the 1st Joint Conference on Lexical and Computational Semantics*, Montréal, Canada, June 2012, p. 335–339.

[20] A. Abu Jbara and D. Radev, "UMichigan: A conditional random field model for resolving the scope of negation," in *Proceedings of the 1st Joint Conference on Lexical and Computational Semantics*, Montréal, Canada, June 2012, p. 328–334.

[21] B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up? sentiment classification using machine learning techniques," in *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing*, Philadelphia, PA, USA, July 2002, p. 79–86.

[22] M. Wiegand, A. Balahur, B. Roth, D. Klakow, and A. Montoyo, "A survey on the role of negation in sentiment analysis," in *Proceedings of the Workshop on Negation and Speculation in Natural Language Processing*, Uppsala, Sweden, July 2010, p. 60–68.

[23] F. Å. Nielsen, "A new ANEW: Evaluation of a word list for sentiment analysis in microblogs," in *Proceedings of the ESWC2011 Workshop on Making Sense of Microposts: Big Things come in Small Packages*, Crete, Greece, May 2011, p. 93–98.

[24] B. Pang and L. Lee, "A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts," in *Proceedings of the 42nd Meeting of the Association for Computational Linguistics*, Barcelona, Spain, July 2004, p. 271–278.

[25] F. Benamara, B. Chardon, T. Mathieu, V. Popescu, and N. Asher, "How do negation and modality impact on opinions?" in *Proceedings of the ACL-2012 Workshop on Extra-Propositional Aspects of Meaning in Computational Linguistics*, Jeju, Republic of Korea, July 2012, p. 10–18.