A Machine-Learning Approach to Negation and Speculation Detection for Sentiment Analysis

Noa P. Cruz
Department of Information Technology, University of Huelva, Huelva, Spain. E-mail: noa.cruz@dti.uhu.es

Maite Taboada
Department of Linguistics, Simon Fraser University, Vancouver, Canada. E-mail: mtaboada@sfu.ca

Ruslan Mitkov
Research Institute for Information and Language Processing, University of Wolverhampton, Wolverhampton, UK. E-mail: R.Mitkov@wlv.ac.uk

Recognizing negative and speculative information is highly relevant for sentiment analysis. This paper presents a machine-learning approach to automatically detect this kind of information in the review domain. The resulting system works in two steps: in the first pass, negation/speculation cues are identified, and in the second phase the full scope of these cues is determined. The system is trained and evaluated on the Simon Fraser University Review corpus, which is extensively used in opinion mining. The results show how the proposed method outstrips the baseline by as much as roughly 20% in the negation cue detection and around 13% in the scope recognition, both in terms of F_1. In speculation, the performance obtained in the cue prediction phase is close to that obtained by a human rater carrying out the same task. In the scope detection, the results are also promising and represent a substantial improvement on the baseline (up by roughly 10%). A detailed error analysis is also provided. The extrinsic evaluation shows that the correct identification of cues and scopes is vital for the task of sentiment analysis.

Introduction

Detecting negative information is essential in most text-mining tasks such as sentiment analysis because negation is one of the most common linguistic means to change polarity. The literature on sentiment analysis and opinion mining in particular has emphasized the need for robust approaches to negation detection, and for rules and heuristics for assessing the impact of negation on evaluative words and phrases, that is, those that convey the author's opinion toward an object, a person, or another opinion. Many authors (e.g., Benamara, Chardon, Mathieu, Popescu, & Asher, 2012; Wiegand, Balahur, Roth, Klakow, & Montoyo, 2010) have shown that this common linguistic construction is highly relevant to sentiment analysis, and that different types of negation have subtle effects on sentiment. In addition, they argue that the automatic study of opinions requires fine-grained linguistic analysis techniques as well as substantial effort in order to extract features for either machine-learning or rule-based systems, so that negation can be appropriately incorporated.

Distinguishing between objective and subjective facts is also crucial for sentiment analysis since speculation tends to correlate with subjectivity. Authors such as Pang and Lee (2004) show that subjectivity detection in the review domain helps to improve polarity classification.

This paper tackles this problem by developing a system to automatically detect both negation and speculation information in review texts, which could help to improve the effectiveness of opinion-mining systems. It works in two phases: in the first pass, the cues are identified and, in the second stage, the full scope (i.e., words in the sentence affected by the keyword) of these cues is determined. The system is trained and evaluated on the Simon Fraser University (SFU) Review corpus (Konstantinova et al., 2012; Taboada, 2008), which is extensively used in opinion mining and consists of 400 product reviews from the website Epinions.com. We are not aware of other approaches that perform the task using this corpus as a learning source and for evaluation purposes.

Applying a support vector machine (SVM) classifier, the proposed method surpasses the baseline by as much as about 20% in the negation cue detection and about 13% in the

Received March 25, 2014; revised February 6, 2015; accepted February 7, 2015

© 2015 ASIS&T • Published online 13 May 2015 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/asi.23533

scope recognition, both in terms of F_1. To the best of our knowledge, this is the first system that addresses speculation in the review domain. The results achieved in the speculation cue detection are close to those obtained by a human rater performing the same task. In the scope detection phase, the results are also promising and they represent a substantial improvement on the baseline (up by roughly 10%). In addition, the extrinsic evaluation demonstrates that the proposed system could improve results in sentiment analysis.

The rest of the paper is organized as follows. First, the most relevant related research is outlined. Second, the proposed machine-learning negation and speculation detection system is presented. The evaluation framework is then detailed and the results are provided and discussed. An error analysis is provided, and the potential of the developed method for addressing sentiment analysis is also shown. The paper finishes with conclusions and future directions.

Related Work

Initial proposals, such as the ones by Polanyi and Zaenen (2006), suggested that a negative item reverses the polarity of the word or phrase it accompanies. This is the approach taken in quite a few papers (Choi & Cardie, 2008; Moilanen & Pulman, 2007; Wilson, Wiebe, & Hoffmann, 2005), also referred to as switch negation (Saurí, 2008). By way of illustration, if the word *good* carried a positive polarity of +3, then *not good* would be assigned −3. However, there are a number of subtleties related to negation that need to be taken into account. One is the fact that there are negators, including *not*, *none*, *nobody*, *never*, and *nothing*, and other words, such as *without* or *lack* (verb and noun), which have an equivalent effect, some of which might occur at a significant distance from the lexical item which they affect. Thus, for adjectives and adverbs, negation is fairly local, although it is sometimes necessary to include, as part of the scope of negation, determiners, copulas, and certain verbs, as we see in Example (1), where negation occurs at a distance from the negated item, that is, from the item in the scope of negation. This includes negation raising, as in (1d), where the negation and the negated element are in different clauses.

(1a) Nobody gives a good performance in this movie (*nobody* negates *good*).

(1b) Out of every one of the 14 tracks, none of them approach being weak and are all stellar (*none* negates *weak*).

(1c) Just a V-5 engine, nothing spectacular. (*nothing* negates *spectacular*).

(1d) I don't think it's very good (*don't* negates *very good*).

Parsing the text would naturally be the best way to adequately deal with negation, so that the scope of the negation can be appropriately identified. This requires, however, a highly accurate and sufficiently deep parser.

Other approaches to negation address the complexities of negative statements by not simply reversing the polarity of the word or phrase in question, but by shifting it. This reflects the fact that negative statements are often not the opposite of the corresponding positive (Horn, 1989). The switch negation method seems to work well in certain cases (Choi & Cardie, 2008), but it has been shown to perform poorly in others (Liu & Seneff, 2009). Consider the Taboada, Brooke, Tofiloski, Voll, and Stede's (2011) approach, where words are classified in a −5 to +5 scale. Negation of *excellent*, which is an adjective with a positive value of +5, would result in *not excellent*, which intuitively is not necessarily extremely negative, that is, not a −5 word. In fact, *not excellent* seems more positive than *not good*, which in Taboada et al.'s dictionary would be assigned a −3. To capture these pragmatic intuitions, another method of negation is proposed, a polarity shift or shift negation (Saurí, 2008; Taboada et al., 2011). Instead of changing the sign, the word's value is shifted toward the opposite polarity by a fixed amount.

Benamara et al. (2012), in the context of sentiment analysis in French, distinguish between different types of negation: negative operators (*not*, *no more*, *without*, or their French equivalents), negative quantifiers (*nobody*, *nothing*, *never*), and lexical negations (*lack*, *absence*, *deficiency*). They show that each type has different effects on both the polarity and the strength of the negation. Specifically, they found that negation always changes the polarity, but that the strength of an opinion expression in the scope of negation is not greater than that of the opinion expression alone. Furthermore, opinions in the scope of multiple negatives have a higher strength than if in the scope of a single negative. Therefore, dealing with negation requires going beyond polarity reversal, since simply reversing the polarity of sentiment upon the appearance of negation may result in inaccurate interpretation of sentiment expressions. Liu and Seneff (2009) put forward a linear additive model that treats negations as modifying adverbs because they also play an important role in determining the degree of the orientation level. For example, *very good* and *good* certainly express different degrees of positive sentiment and *not bad* does not express the opposite meaning of *bad*, which would be highly positive. For that reason, the authors propose an approach to extracting adverb-adjective-noun phrases based on clause structure obtained by parsing sentences into a hierarchical representation. They also provide a robust general solution for modeling the contribution of adverbials and negation to the score for degree of sentiment.

Yessenalina and Cardie (2011) represent each word as a matrix and combine words using iterated matrix multiplication, which allows for modeling both additive (for negations) and multiplicative (for intensifiers) semantic effects. Similar to other authors, they consider that negation affects both the polarity and the strength of an opinion expression.

For their part, Zirn, Niepert, Stuckenschmidt, and Strube (2011) affirm that in the problem of determining the polarity of a text, in most of the cases it is not only necessary to derive the polarity of a text as a whole, but also to extract negative and positive utterances on a more fine-grained level. To address this issue, they developed a fully automatic framework combining multiple sentiment lexicons,

DOI: 10.1002/asi

discourse relations, and neighborhood information (specifically, the polarity of the neighboring segment and the relation between segments because this can help to determine the polarity out of context). Their experiments show that the use of structural features improves the accuracy of polarity predictions, achieving accuracy scores of up to 69%, which significantly outperforms the baseline (51.60%). Polanyi and van der Berg (2011) also discuss the application of the Linguistic Discourse Model (Polanyi, 1986) to sentiment analysis at the discourse level. They focus on movie reviews, because they are characterized by shifting contexts of sentiment source and target. Their approach enables aggregation of different units, based on the type of discourse relation holding between the units. For instance, two units of positive polarity in a coordination relation will join to result in a positive larger unit. A contrast relation, on the other hand, would have to take into account potential different polarities in each of its component units.

Identifying speculative information is also crucial for sentiment analysis because hedging is a linguistic expression that tends to correlate with subjectivity (Montoyo, Martínez-Barco, & Balahur, 2012). As Saurí and Pustejovsky (2009) explain, the same situation can be presented as an unquestionable fact in the world, a mere possibility, or a counterfact according to different sources. For example, in (2a) the author is presenting the information as corresponding to a fact in the world. On the other hand, the author of (2b) is characterizing the information only as a mere possibility.

(2a) The US Soccer Team plays against Spain in October.

(2b) The US Soccer Team may play against Spain in October.

Pang and Lee (2004) propose to employ a subjectivity detector before classifying the polarity. This detector determines whether each sentence is subjective, discarding the objective ones and creating an extract that should better represent a review's subjective content. The results show how subjectivity detection in the review domain helps to improve polarity classification. Wilson et al. (2005) suggest that identification of speculation in reviews can be used for opinion mining since it provides a measure of the reliability of the opinion contained. Recently, authors such as Benamara et al. (2012) have studied the effect of speculation on opinion expressions according to their type (i.e., buletic, epistemic, and deontic). They highlight that, as occurs in negation, each of these types has a specific effect on the opinion expression in its scope and this information should be used as features in a machine-learning setting for sentence-level opinion classification.

As a result of all these studies, and due to the clear need to take into consideration negation and speculation, many authors have developed negation/speculation detection systems that help to improve the performance in natural language processing tasks such as sentiment analysis. A great deal of the work in this regard has been done in the biomedical domain because of the availability of the BioScope corpus, a collection of clinical documents, full papers, and abstracts annotated for negation, speculation, and their scope (Szarvas, Vincze, Farkas, & Csirik, 2008). These approaches evolve from rule-based techniques to machine-learning ones.

Among the rule-based studies, the one developed by Chapman, Bridewell, Hanbury, Cooper, and Buchanan (2001) stands out. Their algorithm, NegEx, determines whether a finding or disease mentioned within narrative medical reports is present or absent. Although the algorithm is described by the authors themselves as simple, it has proven to be powerful in negation detection in discharge summaries. The reported results of NegEx show a positive predictive value (PPV or precision) of 84.5%, sensitivity (or recall) of 77.8%, and a specificity of 94.5%. However, when NegEx is applied to a set of documents from a different domain than that for which it was conceived, the overall precision is lower by about 20 percentage points (Mitchell, 2004). Other interesting research based on regular expressions is the work of Mutalik, Deshpande, and Nadkarni (2001), Elkin et al. (2005), Huang and Lowe (2007), and Apostolova, Tomuro, and Demner-Fushman (2011).

However, most work in the field of negation/speculation detection is based on machine-learning approaches, a notable example of which is the research conducted by Morante and Daelemans (2009b). Their system consists of four classifiers. Three classifiers predict whether a token is the first token, the last token, or neither in the scope sequence. A fourth classifier uses these predictions to determine the scope classes. It shows a high performance in all the subcollections of the BioScope corpus: For clinical documents, the F-score of negation detection is 84.2%, and 70.75% of scopes are correctly identified. For full papers, the F-score is 70.94%, and 41% of scopes are correctly predicted. In the case of abstracts, the F-score is 82.60%, and the percent of scopes correctly classified is 66.07%. They port the system initially designed for negation detection to speculation (Morante & Daelemans, 2009a), showing that the same scope-finding approach can be applied to both negation and hedging. In this case, the F-score of speculation detection for clinical documents is 38.16%, while 26.21% of scopes are correctly identified. For papers, the F-score is 59.66%, and 35.92% of scopes are correctly predicted. The F-score for abstracts is 78.54%, and the percentage of scopes correctly classified is 65.55%.

Another recent work is that developed by Agarwal and Yu (2010), who detect negation/speculation cue phrases and their scope in clinical notes and biological literature from the BioScope corpus using conditional random fields (CRF) as a machine-learning algorithm. However, due to the fact that the corpus partitions and the evaluation measures are different, this system is not comparable with those previously described.

An interesting approach to scope learning for negation is that presented by Zhu, Li, Wang, and Zhou (2010). They formulate it as a simplified shallow semantic parsing problem by regarding the cue as the predicate and mapping its scope into several constituents as the arguments of the

DOI: 10.1002/asi

cue. The results show that this kind of system together with an accurate cue classifier could be appropriate for tackling the task.

Drawing on the BioScope corpus, Velldal, Øvrelid, Read, and Oepen (2012) combine manually crafted rules with machine-learning techniques. Dependency rules are used for all cases where they do not have an available Head-driven Phrase Structure Grammar (HPSG) parser. For the cases where they do, the scope predicted by these rules is included as a feature in a constituent ranker model that automatically learns a discriminative ranking function by choosing subtrees from HPSG-based constituent structures. Although the results obtained by this system can be considered as the state of the art, the combination of novel features together with the classification algorithm chosen in the system developed by Cruz Díaz, Maña López, Vázquez, and Álvarez (2012) improves the results to date for the subcollection of clinical documents.

Finally, Zou, Zhou, and Zhu (2013) propose a novel approach for tree kernel-based scope detection by using the structured syntactic parse information. In addition, they explore a way of selecting compatible attributes for different parts of speech (POS) since features have imbalanced efficiency for scope classification, which is normally affected by the POS. For negation, evaluation on the BioScope corpus reports an F-score of 76.90% in the case of the abstracts subcollection, 61.19% for papers, and 85.31% for clinical documents. For speculation, the system yields F-score values of 84.21% for abstracts, 67.24% for papers, and 72.92% for clinical texts in the scope detection phase (using in all cases as cues those that appear annotated as such in the corpus).

In contrast to the biomedical domain, the impact of negation/speculation detection on sentiment analysis has not been sufficiently investigated, perhaps because standard corpora of reasonable size annotated with this kind of information have become available only recently. However, there are a few approaches like the system described by Jia, Yu, and Meng (2009). They propose a rule-based system that uses information derived from a parse tree. This algorithm computes a candidate scope, which is then pruned by removing those words that do not belong to the scope. Heuristic rules are used to detect the boundaries of the candidate scope. These rules include the use of delimiters (i.e., unambiguous words such as *because*) and conditional word delimiters (i.e., ambiguous words like *for*). There are also defined situations in which a negation cue does not have an associated scope. The authors evaluate the effectiveness of their approach on polarity determination. The first set of experiments involves the accuracy of computing the polarity of a sentence, while the second means the ranking of positively and negatively opinionated documents in the TREC blogosphere collection (Macdonald & Ounis, 2006). In both cases, their system outperforms the other approaches described in the literature. Councill, McDonald, and Velikovich (2010) define a system that can identify exactly the scope of negation in free text. The cues are detected using a lexicon (i.e., a dictionary of 35 negation keywords). A CRF is employed to predict the scope. This classifier incorporates, among others, features from dependency syntax. The approach is trained and evaluated on a product review corpus. It yields an 80.0% F-score and correctly identifies 39.8% of scopes. The authors conclude that, as they expected, performance is improved dramatically by introducing negation scope detection (29.5% for positive sentiment and 11.4% for negative sentiment, both in terms of F-score). Dadvar, Hauff, and de Jong (2011) investigate the problem of determining the polarity of sentiment in movie reviews when negation words occur in the sentence. The authors also observe significant improvement in the classification of the documents after applying negation detection.

In this vein, Lapponi, Read, and Ovrelid (2012) present a state-of-the-art system for negation detection. The heart of the system is the application of CRF models for sequence labeling which makes use of a rich of lexical and syntactic features, together with a fine-grained set of labels that capture the scopal behavior of tokens. At the same time, they demonstrate that the choice of representation has a significant effect on the performance.

The annotated corpus used by Councill et al. (2010) and Lapponi et al. (2012) is rather small in size, containing only 2,111 sentences in total. A large-scale corpus is needed for training statistical algorithms to identify these aspects of the language so the use of a bigger annotated corpus such as the SFU Review corpus (which contains 17,263 sentences) could enable the improvement of negation recognition in this domain. In addition, although it has proven that speculation has an effect on the opinion expression and it should be taken into account (Benamara et al., 2012; Pang & Lee, 2004; Wiebe, Wilson, & Cardie, 2005), there is, as far as we are aware, no work on detecting the speculation in the review domain. However, the annotation of the SFU Review corpus with speculative information will make it possible to tackle this problem efficiently.

This paper fills these gaps through the development of a system that makes use of machine-learning techniques to identify both negation and speculation cues and their scope. This system is also novel in that it uses the SFU Review corpus as a learning source and for evaluation purposes. As a result, the proposed system will help to improve the effectiveness of sentiment analysis and opinion-mining tasks.

Method

The *identification of negation and speculation cues* and the *determination of their scope* are modeled as two consecutive classification tasks (see Figure 1). They are implemented using supervised machine-learning methods trained on the SFU Review corpus (Konstantinova et al., 2012).[1]

In the first phase, when the cues are detected, a classifier predicts whether each word in a sentence is the first

[1] See http://www.sfu.ca/~mtaboada/research/SFU_Review_Corpus.html

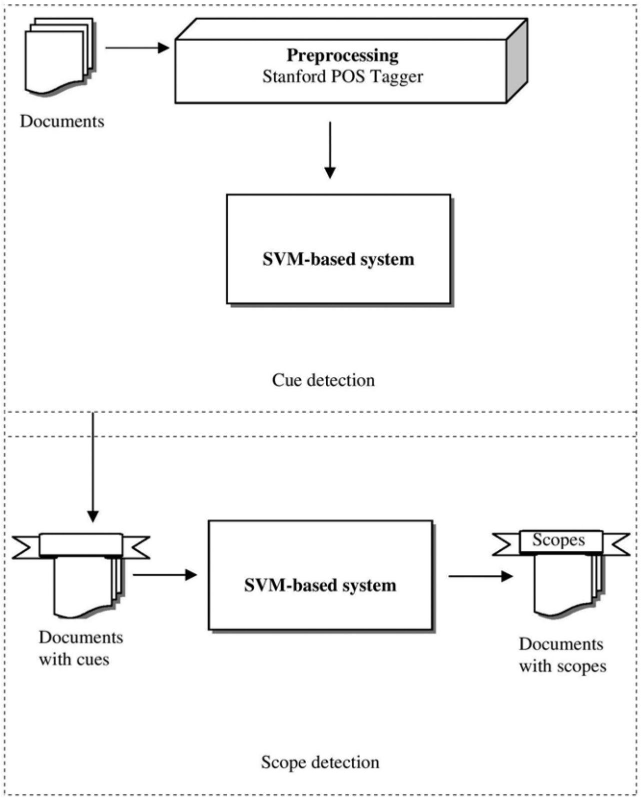


FIG. 1. System architecture.

one of a cue (B), inside a cue (I), or outside of it (O) using a BIO representation. This allows the classifier to find multiword cues (MWCs), which represent, in the SFU Review corpus, 25.80% of the total number of cues in negation and 2.81% in the case of the speculation. For example, in sentence (3), the token *ca* is assigned to the B class; *n't* is tagged as I, and the rest of the tokens in the sentence are tagged as O class.

(3) Murphy Lee raps about him and how women **can't** get enough of him.

In the second step, another classifier decides, at the sentence level, which words are affected by the cues identified in the previous phase. This means determining, for every sentence that has cues, whether the other words in the sentence are inside (IS) or outside (O) the scope of the cue. This process is repeated as many times as there are cues in the sentence. In example (3) the classifier tags the words *enough of him* as IS class, whereas it assigns the class O to the rest of tokens.

The classifiers are trained using a support vector machine (SVM) as implemented in LIBSVM (Chang & Lin, 2011), since it has proved to be powerful in text classification tasks where it often achieves the best performance (Sebastiani, 2002). As kernel, the Radial Basic Function (RBF) is used because previous work (e.g., Cruz Díaz et al., 2012) shows its effectiveness in this task. In addition, the classifier is parameterized optimizing the parameters *gamma* and *cost* using the values recommended by Hsu, Chang, and Lin (2003).

This is a classification problem of imbalanced data sets in which the classification algorithms are biased toward the majority class. To solve this issue, an algorithmic level solution has been considered, that is, cost sensitive learning (CSL) (Kumar & Sheshadri, 2012). The purpose of CSL is usually to build a model with total minimum misclassification costs. This approach applies different cost matrices that describe the cost for misclassifying examples; the cost of misclassifying a minority-class example is substantially greater than the cost of misclassifying a majority-class example (He & Garcia, 2009; He & Ma, 2013). As authors such as Cao, Zaiane and Zhao (2014) explain, assigning distinct costs to the training examples seems to be the most effective approach to class-imbalanced data problems. The cost-sensitive SVM algorithm (CS-SVM) incorporated in the LIBSVM package has been added as an additional benchmark using the *weight* parameter to control the skew of the SVM optimization (i.e., classes with a higher weight will count more).

There have also been experiments with a Naïve Bayes algorithm implemented in Weka (Witten & Frank, 2005), but as shown in the Results, it produces lower results.

Text Collection

The system presented in this paper uses the SFU Review corpus (Taboada, 2008), as a learning source and for evaluation purposes. This corpus is extensively used in opinion mining (Martınez-Cámara, Martın-Valdivia, Molina-González, & Urena-López, 2013; Rushdi Saleh, Martín-Valdivia, Montejo-Ráez, & Ureña-López, 2011; Taboada et al., 2011) and consists of 400 documents (50 of each type) of movie, book, and consumer product reviews from the website Epinions.com. The corpus has several annotated versions (e.g., for appraisal and rhetorical relations), including one where all 400 documents are annotated at the token level with negative and speculative cues, and at the sentence level with their linguistic scope (Konstantinova et al., 2012). The annotation indicates the boundaries of the scope and the cues, as shown in (4) below. In the annotation, scopes are extended to the largest syntactic unit possible and the cues are never included in the scope.

(4) Why <cue ID=“0”type=“speculation”>would </cue><xcope ID=“2”> anyone want to buy this car </xcope> ?

In addition, there are cues without any associated scope. In negation, the number of cues without scope is 192 (5.44% of the total of cues), whereas in speculation there are 248 cues whose scope is not indicated (4.62% of the total of cues).

TABLE 1. Statistics about the SFU Review corpus.

	#Documents	#Sentences	#Words	Avg. length documents (in sentences)	Avg. length documents (in words)	Avg. length sentences (in words)
Books	50	1,596	32,908	31.92	658.16	20.62
Cars	50	3,027	58,481	60.54	1,169.62	19.32
Computers	50	3,036	51,668	60.72	1,033.36	17.02
Cookware	50	1,504	27,323	30.08	546.46	18.17
Hotels	50	2,129	40,344	42.58	806.88	18.95
Movies	50	1,802	38,507	36.04	770.14	21.37
Music	50	3,110	54,058	62.2	1,081.16	17.38
Phones	50	1,059	18,828	21.18	376.56	17.78
Total	**400**	**17,263**	**322,117**	**43.16**	**805.29**	**18.66**

Avg. = average.

TABLE 2. Negation statistics in the SFU Review corpus.

	Books	Cars	Computers	Cookware	Hotels	Movies	Music	Phones	Total
#Negation sentences	362	517	522	320	347	427	418	206	3,119
%Negation sentences	22.7	17.1	17.2	21.3	16.3	23.7	13.4	19.5	**18.1**
#Negation cues	406	576	590	376	387	490	470	232	**3,527**
#Words in scope	2,139	2,939	3,106	1,944	2,038	2,537	3,019	1,146	18,868
#Scope	387	545	570	355	370	445	440	221	3,333
Avg. length scope	5.53	5.39	5.45	5.48	5.51	5.70	6.86	5.19	**5.66**
#Words scope left	12	20	17	20	21	9	8	7	114
#Scope left	6	3	6	3	6	3	2	2	31
Avg. length scope to the left	2	6.67	2.83	6.67	3.50	3.00	4.00	0	3.68
#Words scope right	2,127	2,919	3,089	1,924	2,017	2,528	3,011	1,139	18,754
#Scope right	383	542	568	352	367	442	438	221	3,313
Avg. length scope to the right	5.55	5.39	5.44	5.47	5.50	5.72	6.87	5.15	5.66
% Scope to the left	1.55	0.55	1.05	0.85	1.62	0.67	0.45	0.90	0.93
% Scope to the right	98.97	99.45	99.65	99.15	99.19	99.33	99.55	100.00	99.40

Avg. = average.
Avg. length of scope is shown in number of words.
A word is counted as many times as it appears in scope.
There are scopes which extend to the left and the right of the cue, so we count them twice (once as *#Scope left* and again as *#Scope right*).

Table 1 summarizes the main characteristics of the SFU Review corpus. As the third column shows, the number of sentences of the corpus is 17,263. It is of considerable size, especially compared to the only other available corpus in the review domain described in Councill et al. (2010), which contains 2,111 sentences in total. Furthermore, the corpus by Councill et al. was annotated only for negation, but not speculation. The SFU Review corpus is also larger than other corpora of different domains like the ConanDoyle-neg corpus (consisting of 4,423 sentences annotated with negation cues and their scope) and comparable in size to BioScope, which contains slightly more than 20,000 annotated sentences altogether. Another well-known corpus in this domain is the FactBank (Saurí & Pustejovsky, 2009). It consists of 208 documents from newswire and broadcast news reports annotated with factual information. However, the annotation was done at the event level so it cannot be compared to the SFU Review corpus.

In the case of negation, out of the total number of 17,263 sentences, 18% contained negation cues,[2] as shown in Table 2. However, this proportion varies slightly depending on the domain. Negation is even more relevant in this corpus than in others like the BioScope corpus where 13% of the sentences contain negations. This highlights the importance of negation resolution to sentiment analysis.

In the case of speculation, as Table 3 shows, 22.7% of all sentences are speculative.[3] This proportion is higher than the negative sentences because of the nature of the corpus,

[2] The most frequent negation cues are *not* (40.23%) and *no* (14.85%). They constitute more than 55% of the total frequency of all the negation cues found in the corpus.

[3] *If* (16.34%), *or* (15.30%), and *can* (14.27%) are some of the most frequent speculation cues. They do not represent the majority of speculation cases since the number of occurrences of each cue was equally distributed across all the documents.

TABLE 3. Speculation statistics in the SFU Review corpus.

	Books	Cars	Computers	Cookware	Hotels	Movies	Music	Phones	Total
#Speculation sentences	275	788	704	411	505	469	470	290	3,912
%Speculation sentences	17.2	26.0	23.2	27.3	23.7	26.0	15.1	27.4	**22.7**
#Speculation cues	370	1,068	944	583	695	648	643	408	**5,359**
#Words in scope	2,791	7,738	6,567	4,048	4,582	4,770	5,433	2,889	38,818
#Scope	360	1,036	919	545	655	615	608	387	5,125
Avg. length scope	7.75	7.47	7.15	7.43	7.00	7.76	8.94	7.47	**7.57**
#Words scope left	217	554	462	505	407	315	341	149	2,950
#Scope left	66	191	153	120	128	97	88	56	899
Avg. length scope to the left	3	0.00	3.02	0.00	0.00	3.25	3.88	2.66	3.28
#Words scope right	2,574	7,184	6,105	3,543	4,175	4,455	5,092	2,740	35,868
#Scope right	359	1,036	917	544	655	611	605	387	5,114
Avg. length scope to the right	7.17	6.93	6.66	6.51	6.37	7.29	8.42	7.08	7.01
% Scope to the left	18.33	18.44	16.65	22.02	19.54	15.77	14.47	14.47	17.54
% Scope to the right	99.72	100.00	99.78	99.82	100.00	99.35	99.51	100.00	99.79

Note. Same notes as in Table 2 apply.

where speculation is widely used to express opinions. By comparison, less than 20% of the sentences in the BioScope corpus are speculative.

Attributes

All tokens that appear in the collection of documents used for the experimentation are represented by a set of features that are different in each of the two phases into which the task is divided. It has been started by building a pool of baseline features for the classifier based on experience and previous work such as Morante and Daelemans (2009b) and Cruz Díaz et al. (2012) (i.e., lemma and POS of the token in focus as well as whether it is at the beginning or end of the sentence for the cue detection; lemma and POS of the cue, token in focus, and one token on both the left and right of the token in focus in the scope detection). As features have an imbalanced classification, a greedy forward procedure to obtain the final feature set was followed. It consists of adding a specialized new feature outside the basic set and removing a feature inside it, one by one, in order to check how each feature contributes to improving the performance. This procedure is repeated until no feature is added or removed, or the performance does not improve.

In the cue detection phase, instances represent all tokens in the corpus. As many authors such as Øvrelid, Velldal, and Oepen (2010) suggest, syntactic features seem unnecessary, since cues depend on the token itself and not the context. Therefore, lexical information is the key in this phase, which is why token-specific features have been used; these are detailed in Table 4.

TABLE 4. Features in the cue detection phase.

Feature name	Description
	Token-level features
Lemma$_i$	Lemma of token in focus
POS$_i$	Part-of-speech of token in focus
Begin sentence$_i$	Boolean tag to indicate if the token is the first token in the sentence
End sentence$_i$	Boolean tag to indicate if the token is the last token in the sentence
	Contextual features
Lemma$_{i-1}$	Lemma of token$_{i-1}$
POS$_{i-1}$	Part-of-speech of token$_{i-1}$
Begin sentence$_{i-1}$	Boolean tag to indicate if token$_{i-1}$ is the first token in the sentence
End sentence$_{i-1}$	Boolean tag to indicate if token$_{i-1}$ is the last token in the sentence
Lemma$_{i+1}$	Lemma of token$_{i+1}$
POS$_{i+1}$	Part-of-speech of token$_{i+1}$
Begin sentence$_{i+1}$	Boolean tag to indicate if token$_{i+1}$ is the first token in the sentence
End sentence$_{i+1}$	Boolean tag to indicate if token$_{i+1}$ is the last token in the sentence

Note. Part-of-speech tags are returned by the Stanford POS tagger.[4]

Feature selection experiments reveal that the most informative features in this phase are the *lemma of the token*, followed by the *lemmas of the neighboring words* in the case of negation. For speculation, the most important information is the *lemma of the token* and its *POS*.

In the scope detection phase, an instance represents a pair of a cue and a token from the sentence. This means that all tokens in a sentence are paired with all negation or speculation cues that occur in the sentence. Table 5 includes the features that directly relate to the characteristics of cues or tokens and their context used in this phase.

Besides the feature set listed previously, syntactic features between the token in focus and cues are explored in the classifier because previous research has shown that highly accurate extraction of syntactic structure is beneficial for the scope detection task. For example, Szarvas et al. (2008) point out that the scope of a keyword can be determined on the basis of syntax (e.g., the syntactic path from the token to the cue, its dependency relation, etc.), and Huang and Lowe

[4] http://nlp.stanford.edu/software/tagger.shtml

TABLE 5. Features in the scope detection phase.

Feature name	Description
	About the cue
Lemma	Lemma of the cue
POS	Part-of-speech of the cue
About the paired token	
Lemma	Lemma of paired token
POS	Part-of-speech of paired token
Location	Location of the paired token in relation to the cue (before, inside, or after the cue)
	Tokens between the cue and the token in focus
Distance	Distance in number of tokens between the cue and the token in focus
Chain-POS	Chain of part-of-speech tags between the cue and the token in focus
Chain-Types	Chain of types between the cue and the token in focus
Other features	
Lemma$_{i-1}$	Lemma of token to the left of token in focus
Lemma$_{i+1}$	Lemma of token to the right of token in focus
POS$_{i-1}$	Part-of-speech of token to the left of token focus
POS$_{i+1}$	Part-of-speech of token to the right of token focus
Place cue	Place of the cue in the sentence (position of the cue divided by the number of tokens in the sentence)
Place token	Place of the token in focus in the sentence (position of the token in focus divided by the number of tokens in the sentence)
	Dependency syntactic features
Dependency relation	Kind of dependency relation between the token in focus and the cue
Dependency direction	If the token in focus is head or dependent
POS first head	Part-of-speech of the first order syntactic head of token in focus
POS second head	Part-of-speech of the second order syntactic head of token in focus
Token ancestor cue	Whether the token in focus is ancestor of the cue
Cue ancestor token	Whether the cue is ancestor of the token in focus
Short path	Dependency syntactic shortest path from the token in focus to the cue
Dependency graph path	Dependency syntactic shortest path from the token in focus to the cue encoding both the dependency relations and the direction of the arc that is traversed
Critical path	Dependency syntactic shortest path from the cue to the token in focus
Number nodes	Number of dependency relations that must be traversed in the short path

Note. Part-of-speech tags are returned by the Stanford POS tagger (See footnote 4).

(2007) note that structure information stored in parse trees helps to identify the scope of negative hedge cues. Both constituent and dependency syntactic features have been shown to be effective in scope detection (Özgür & Radev, 2009). In 1965, Gaifman proved that dependency and constituency grammars are strongly equivalent. More recently, other authors such as Ballesteros Martínez (2010) also affirmed that both type of analysis are equivalents. In fact, an automatic method to transform a constituent tree into a dependency one exists (Gelbukh, Torres, & Calvo, 2005). Dependency representations were opted for because they are more compact than constituent structures since the number of nodes is constrained by the number of tokens of the sentence. This kind of information can be provided by Maltparser (Nivre, Hall, & Nilsson, 2006), a data-driven dependency parser.

Drawing upon the research so far which examines the relationship between cues and tokens by dependency arcs in the negation and speculation scope detection task (Councill et al., 2010; Lapponi et al., 2012; Zou et al., 2013), the final rows of Table 5 show the proposal for an operational set of syntactic features.

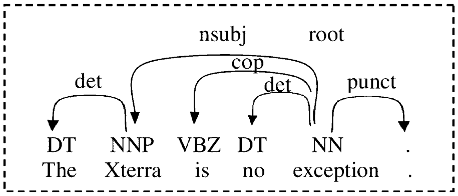


FIG. 2. Example dependency graph.

Figure 2 is an illustration of the corresponding dependency tree of the sentence “The Xterra is no exception.” In this example, if the token *the* is taken to be the token in focus to determine whether it is inside the scope of the cue *no*, then the features *POS first head* and *POS second head* have the value *NNP* and *NN*, respectively. The cue is an ancestor of the token, so the token is not an ancestor of the cue. The short path is formed by the dependencies *det nsubj det* and the number of the nodes that must be traversed from one node to another is 3, since we take into account the cue and the token itself. The *critical path* in this case is the same as the *short path*. In addition, the concept of *dependency graph path* used in Lapponi et al. (2012) and first introduced by Gildea and Jurafsky (2002) was employed as a feature. It represents the shortest path traversed from the token in focus to the cue, encoding both the dependency relations and the direction of the arc being traversed. For instance, as

DOI: 10.1002/asi

described in Figure 2, (5) shows the dependency graph path between *the* (token in focus) and *no* (cue).

(5) *det* ↑ *nsubj* ↓ *det*.

Finally, feature selection experiments show that the most informative features for both negation and speculation in this phase are the *chain of part-of-speech tags* between the cue and the token in focus, followed by the *dependency graph path*, *critical path*, and *short path*.

Evaluation Measures

The standard measures precision (P), recall (R), and their harmonic mean F_1-score (Rijsbergen, 1979) are used to assess the performance in terms of both cue and scope detection, since this is the evaluation scheme followed by all the authors in this task (e.g., Councill et al., 2010; Lapponi et al., 2012; Morante & Daelemans, 2009a, 2009b) and employed in the different shared tasks and competitions related to this topic (e.g., the CoNLL-2010 Shared Task [Farkas, Vincze, Móra, Csirik, & Szarvas, 2010] or the SEM 2012 Shared Task [Morante & Blanco, 2012]). In addition, F_1-score is a well-established metric suited to imbalanced data sets (He & Ma, 2013).

Precision accounts for the reliability of the system's predictions, recall is indicative of the system's robustness, while F_1-score quantifies its overall performance.

In the cue detection task, a token is correctly identified if its position has been accurately determined to be at the beginning, inside, or outside the cue. Precision and recall can be calculated as follows:

$$P = \frac{\text{\# tokens correctly negated by the system}}{\text{\# tokens negated by the system}}$$

$$R = \frac{\text{\# tokens correctly negated by the system}}{\text{\# tokens negated in the test collection}}$$

In the task of detecting the scope, a token is correctly classified if it is properly identified as being inside or outside the scope of each of the cues that appear in the sentence. In this case, precision and recall are computed as follows:

$$P = \frac{\begin{array}{c}\text{\# tokens belonging to some scope correctly}\\ \text{detected by the system}\end{array}}{\begin{array}{c}\text{\# tokens belonging to some scope}\\ \text{detected by the system}\end{array}}$$

$$R = \frac{\begin{array}{c}\text{\# tokens belonging to some scope correctly}\\ \text{detected by the system}\end{array}}{\begin{array}{c}\text{\# tokens belonging to some scope in}\\ \text{the test collection}\end{array}}$$

In both cases, $F_1 = \frac{2PR}{P+R}$.

Although the F_1-score is very popular and suitable for dealing with the class-imbalance problem, it is focused on the positive class only. Therefore, the Geometric Mean (G-mean) has been used as an additional measure since it takes into account the relative balance of the classifier's performance on both the positive and the negative classes (He & Ma, 2013). It is a good indicator of overall performance (Cao et al., 2014), and has been employed by several researchers for evaluating classifiers on imbalanced data sets (Akbani, Kwek, & Japkowicz, 2004; Barua, Islam, Yao, & Murase, 2014).

G-mean is calculated as √ sensitivity*specificity, where sensitivity = R and specificity corresponds to the proportion of negative examples that are detected by the system.

In the scope detection task, and following previous research, the percentage of scopes correctly classified has been also measured. Specifically, two different definitions are adopted, which have been used by other authors for the same task. First, the measure proposed by Morante and Daelemans (2009a, 2009b) has been employed, where a scope is correct if all the tokens in the sentence have been correctly classified as inside or outside the scope of the cue (percentage of correct scopes, henceforth PCS). It can be considered a strict way to evaluate scope resolution systems. Second, the more relaxed approach put forward by Councill et al. (2010) in which the percentage of correct scopes is calculated as the number of correct spans divided by the number of true spans (percentage of correct relaxed scopes, from now on PCRS) has been applied. Therefore, in this case, a scope is correct simply if the tokens in the scope have been correctly classified as inside of it.

The evaluation in terms of precision and recall measures considers a token as a unit, whereas the evaluation in terms of PCS and PCRS regards a scope as a unit. It should be noted that negation and speculation detection are evaluated separately.

Finally, a two-tailed sign test applied to the token-level predictions was used with the aim of assessing the statistical significance of differences in performance. This is the simplest nonparametric test for matched or paired data that, in this case, will compare the differences in the prediction of two given classifiers. A significance level of $\alpha = 0.05$ was assumed.

Results

The results reported in this section were obtained by employing 10-fold cross-validation. For each fold, a document-level partitioning of the data was used, randomly selecting as well as balancing the number of documents in each of these folds.

As detailed in the Method section, experiments were undertaken with Naïve Bayes and SVM classifiers. Simple baselines models were also used in both phases to compare the results. The following sections detail the results for the cue and scope detection tasks.

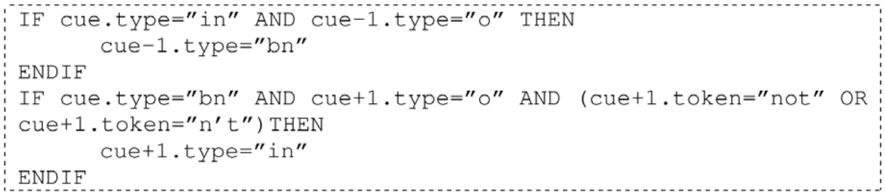
Cue Detection Results

Table 6 shows the results for negation and speculation cue detection.

TABLE 6. Results for detecting negation and speculation cues: Averaged 10-fold cross-validation results for the baseline algorithm and both Naïve Bayes and SVM classifiers on the SFU Review corpus training data. Results are shown in terms of Precision, Recall, F_1, and G-mean (%).

		Negation				Speculation			
Model		Prec	Rec	F_1	G-mean	Prec	Rec	F_1	G-mean
Stratified	Baseline	93.54	55.08	69.34	74.20	91.54	57.00	70.26	75.46
	Naïve Bayes	63.26 (65.91)	68.95 (73.15)	65.98 (69.34)	82.54 (85.33)	72.05	75.05	73.52	86.42
	SVM RBF	82.44 (89.64)	93.22 (95.63)	87.50 (**89.64**)	96.44 (97.69)	90.73	93.97	92.32	96.86
	CS-SVM	80.40	97.86	88.28	98.79	88.03	96.36	92.00	98.05
Random	Naïve Bayes	63.22 (65.65)	68.72 (72.52)	65.86 (68.92)	82.71 (84.99)	72.03	74.69	73.34	86.21
	SVM RBF	82.67 (84.30)	93.47 (95.52)	87.74 (89.56)	96.57 (97.63)	90.74	94.06	**92.37**	**96.90**
	CS-SVM	80.49	97.84	88.32	98.78	88.06	96.37	**92.03**	**98.06**

SVM = Support Vector Machine; RBF = Radial Basic Function kernel; Prec = Precision; Rec = Recall; CS = Cost-Sensitive Learning.
In parentheses, results obtained after applying the postprocessing algorithm.
Results obtained by CS-SVM after applying the postprocessing algorithm are not shown because they are the same as without applying it. The same occurs with all the speculation detection approaches.
Note that "Random" means the #documents in each fold of the cross-validation are randomly selected, whereas in "Stratified" the #documents is the same in all the folds.



```
IF cue.type="in" AND cue-1.type="o" THEN
      cue-1.type="bn"
ENDIF
IF cue.type="bn" AND cue+1.type="o" AND (cue+1.token="not" OR
cue+1.token="n't")THEN
      cue+1.type="in"
ENDIF
```

FIG. 3. Cue detection postprocessing algorithm pseudocode.

A simple postprocessing algorithm was applied to the output of the classifier in order to reduce the cases of failure to detect the most common type of multiword cues (MWCs) that appears in the SFU Review corpus (i.e., MWCs formed by two words, the last one being *n't* or *not*). The postprocessing algorithm works as follows: If a word is identified at the beginning of a cue and the following word is identified as being outside it but the word is *n't* or *not*, the algorithm changes the type of this final word to being inside the cue. In addition, if a token is classified as being inside of a cue and its predecessor word is classified as outside, it changes the class of this final token to the start of a cue. Figure 3 shows the pseudocode of this algorithm.

This postprocessing is very effective in negation because the percentage of MWCs is 25.80%. In speculation, 2.81% of MWCs cause the algorithm not to be effective in this case.

Although the results obtained in the speculation detection task are by and large slightly higher than those achieved in negation detection, all the algorithms performed satisfactorily. In addition, no large differences were observed between performing the cross-validation randomly or in a stratified way.

Baseline results are shown in the third row of Table 6. It has been created by tagging as cue the most frequent negation and speculation expressions that appear in the training data set (i.e., those that cover more than 50% of the total number of cues). To achieve the baseline, the two most frequent expressions for negation (i.e., *no* and *not*) and the four most frequent expressions for speculation (i.e., *if*, *or*, *can*, and *would*) are used because in this case the most frequent expressions are not concentrated in a small number of cues as occurs for negation.

This baseline proves to be competitive in precision where it actually outperforms all the other systems. In terms of F_1, the results are improvable for both negation (69.34%) and speculation (70.26%). Furthermore, the results yielded by the baseline in the negation detection are comparable with those obtained by Naïve Bayes (the latter achieves an F_1 of 68.92% using the random-selection option and 69.34% in the stratified way, both after applying postprocessing). In the case of speculation, as shown in the last column, Naïve Bayes shows a slight improvement on the baseline (73.34% or 73.52%, depending on the way the documents are selected in the cross-validation), this difference being statistically significant according to a two-tailed sign-test ($p = .0009$). In terms of G-mean, Naïve Bayes also outstrips the baseline by about 10% (both in negation and speculation). However, these two approaches appear to have somewhat different strengths and weaknesses. The Naïve Bayes classifier shows higher recall, whereas, as mentioned before, the baseline is stronger in terms of precision.

The best F_1 and G-mean for both negation and speculation is obtained by the SVM classifier. The cost-sensitive learning applied to SVM slightly improves the results in terms of G-mean. However, it does not happen the same in terms of F_1 (the measure used for all the authors in this task to assess the performance of their systems). This is due to different factors. First, the precision shown by the cost-sensitive learning approach is low since the classifier introduces many false-positive errors trying to minimize the cost

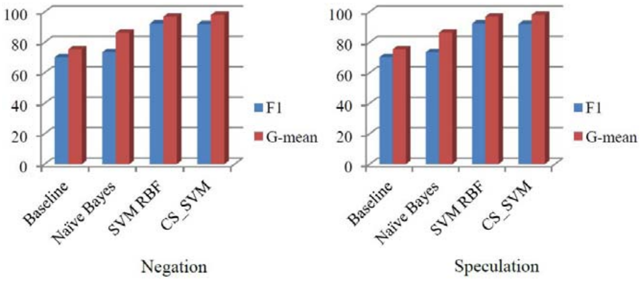


FIG. 4. Comparison of the results obtained by the different approaches in the cue detection task in terms of F_1 and G-mean (%). [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

function (the cost for misclassifying any example belonging to the majority class is small). Next, the postprocessing algorithm is not effective in negation detection because most errors are derived from the fact that the classifier identifies as cues words ones that are not annotated as such in the corpus (false-positive errors) and not as a result of an incorrect classification of MWCs. Finally, an SVM classifier without any modifications seems sufficient to solve this problem since it performs well with moderately imbalanced data (Akbani et al., 2004), as is the case here.

In speculation, the results obtained by the SVM classifier represent a substantial improvement on the baseline (up by roughly 22%). It also outstrips the Naïve Bayes results by 20% in terms of F_1 and 10% according to G-mean (see Figure 4). As shown by the two-tailed sign test, these differences ($p = 9.33^{E-17}$ compared to the baseline; $p = 1.69^{E-14}$ if it is compared to Naïve Bayes) are significant. The interannotator agreement rates may offer some further perspective on the results discussed here. When creating the SFU corpus, a first annotator annotated the whole corpus. Another expert annotator worked with 10% of the documents from the original collection (randomly selected), annotating them according to the guidelines used by the first annotator. The agreement rate between the second annotator and the chief annotator is 89.12% and 89% in F_1 and kappa measures, respectively. This suggests that the results could be compared with those obtained by an annotator doing the same task.

Negation detection is more complicated. Although the most frequent negation cues are concentrated in a small number of expressions (*no* and *not* represent 55.03% of the total number of cues), what makes negation detection difficult is the large number of MWCs present in the corpus (25.80%). This does not occur in speculation, where the percentage of MWCs is just 2.81%. The results improve with postprocessing, nearing those obtained when identifying speculation. A two-tailed sign-test shows that there is a statistically significant difference between the SVM results before and after applying the postprocessing algorithm ($p = .0013$).

Overall, the results for negation are competitive. In fact, the SVM classifier outperforms the baseline results by as much as about 20% both in terms of F_1 and G-mean and independently of the way in which the cross-validation is done. These differences are deemed significant ($p = 4.47^{E-13}$). Comparing with Naïve Bayes, the proposed method outstrips it by up 20% in terms of F_1 and 12% in terms of G-mean as can be seen in Figure 4. The differences are also significant ($p = 1.33^{E-14}$). In addition, the interannotator agreement rates for negation cues (F_1 of 92.79% and kappa value of 92.7%) in the SFU Review corpus are close to those obtained by a human rater performing the same task.

TABLE 7. Results for detecting negation and speculation scopes with gold standard cues: Averaged 10-fold cross-validation results for the **baseline** algorithm on the SFU Review corpus training data. Results are shown in terms of Precision, Recall, F_1, G-mean, PCS, and PCRS (%).

	Precision	Recall	F_1	G-M	PCS	PCRS
Negation	78.80	66.21	71.96	80.92	23.07	58.03
Speculation	71.77	65.68	68.59	79.75	13.86	45.49

G-M = G-mean; PCS = Percentage of Correct Scopes (all the tokens in the sentence have been correctly classified); PCRS = Percentage of Correct Relaxed Scopes (all the tokens in the scope have been correctly classified).

Finally, it is worth noting that a factor that may have slightly deflated the results, as authors like Velldal et al. (2012) point out, is the use of a document-level rather than a sentence-level partitioning of the data for cross-validation since the latter favors that the number of cues in each fold is more balanced, facilitating, therefore, the detection.

Scope Detection Results

This section presents the results of the scope detection for both the gold standard cues as well as the predicted ones. First, in order to isolate the performance of the scope recognition, the set of cues that appear annotated as such in the SFU Review corpus was used. Next, to measure the performance of the whole system the best scope detection approach was assessed using the cues identified by the classifier in the previous phase.

Tables 7–9 detail the results for the gold standard cues. In general, they show how difficult the task of identifying the scope is compared to the task of recognizing the cues. In addition, in contrast to cue detection, the results for speculation are lower than those obtained by negation. This can be explained by the fact that speculation leads to a text with a greater degree of complexity (e.g., the number of scopes is higher, the average length of the scopes in number of words is longer, as shown in Tables 2 and 3).

Different sets of features were used for both Naïve Bayes and SVM, which aim to show how syntactic information improves the classifier performance. First, a basic configuration consisting of the lemma and POS of the cue, token in focus, and one token on both to the left and right of the token in focus. Next, fine-grained features related to the cue, the token itself, and the context were added. The last configuration also includes the set of syntactic attributes described in Table 5.

In addition, the results were compared with a baseline. This was proposed as a result of the analysis carried out by

TABLE 8. Results for detecting negation and speculation scopes with gold standard cues: Averaged 10-fold cross-validation results for **Naïve Bayes** classifier on the SFU Review corpus training data. Results are shown in terms of Precision, Recall, F_1, G-mean, PCS, and PCRS (%).

		Random						Stratified					
	Configuration (features)	Prec	Rec	F_1	G-M	PCS	PCRS	Prec	Rec	F_1	G-M	PCS	PCRS
Negation	Baseline	47.56	43.12	45.23	64.70	8.02	33.22	47.48	41.22	44.13	63.30	7.93	31.89
	Contextual	76.60	77.79	77.19	87.55	**41.13**	73.15	76.51	78.33	**77.41**	87.85	40.60	74.17
	Dependency syntactic	72.35	80.53	76.22	88.88	38.95	71.78	72.58	81.14	76.62	**89.23**	38.30	**77.71**
Speculation	Baseline	28.00	35.06	31.14	55.93	3.04	18.90	28.56	34.23	31.14	55.43	2.70	18.43
	Contextual	37.96	66.14	48.24	75.90	19.20	59.76	39.41	70.23	**50.49**	78.20	**19.33**	61.00
	Dependency syntactic	35.84	68.27	47.09	76.35	18.28	56.57	36.64	72.08	48.67	**78.34**	18.52	**64.30**

Prec = Precision; Rec = Recall; G-M = G-mean; PCS = Percentage of Correct Scopes (all the tokens in the sentence have been correctly classified); PCRS = Percentage of Correct Relaxed Scopes (all the tokens in the scope have been correctly classified).

TABLE 9. Results for detecting negation and speculation scopes with gold standard cues: Averaged 10-fold cross-validation results for **SVM** classifier on the SFU Review corpus training data. Results are shown in terms of Precision, Recall, F_1, G-mean, PCS, and PCRS (%).

		Random						Stratified					
	Configuration (features)	Prec	Rec	F_1	G-M	PCS	PCRS	Prec	Rec	F_1	G-M	PCS	PCRS
Negation	Baseline	59.79	38.20	46.62	61.32	10,88	29,08	59.52	37.86	46.28	61.04	10.88	28.94
	Contextual	84.02	80.61	82.28	89.36	53.58	77.43	83.29	80.38	81.81	89.21	52.90	77.17
	Dependency syntactic	85.92	81.67	83.74	90.05	57.64	78.84	85.91	81.87	83.84	90.11	57.86	79.13
	Dependency syntactic CS	85.59	82.28	83.91	90.32	58.23	80.14	85.56	82.64	**84.07**	**90.42**	**58.69**	**80.26**
Speculation	Baseline	49.49	36.75	41.18	59.25	4,62	19,20	49.29	36.04	41.64	58.69	4.29	19.91
	Contextual	77.79	75.97	76.87	86.11	39.61	68.10	77.41	75.69	76.54	85.84	37.86	66.71
	Dependency syntactic	79.47	77.01	78.22	86.70	43.04	69.62	79.91	77.32	78.59	86.90	43.90	69.69
	Dependency syntactic CS	79.07	77.77	78.41	87.09	43.40	71.17	79.98	77.80	**78.88**	**87.14**	**43.94**	**71.43**

Note. Same notes as in Table 8 apply. CS = Cost-Sensitive Learning. Optimized values of the parameters c and g: c = 32; g = 0.03125.

Hogenboom, van Iterson, Heerschop, Frasincar, and Kaymak (2011) on a set of English movie review sentences. In that study, the authors show that the best approach to determining the scope of a negation cue is to consider a fixed window length of words following the negation keyword. In the SFU review corpus, the proportion of scopes to the left of the negation cues is virtually nonexistent (0.93%). In contrast, 99.40% of the scopes extend to the right of the cue with an average length of 5.66 words. Therefore, the baseline was created by tagging as scope five words to the right of the cue. In the case of speculation, almost all of the scopes are to the right of the cue (99.79%), with their average length being 7.01 words. The proportion of scopes to the left of the cue is higher than in negation (7.01%), with an average length of 3.28 words. However, the baseline just includes seven words to the right of the cue as inside the scope, since adding information about the left scopes, as Hogenboom et al. (2011) affirm, produces lower results.

This baseline, as shown in the fourth column of Table 7, achieves a promising performance value in terms of F_1 (71.96% for negation and 68.59% for speculation) and G-mean (80.92% and 79.75% for negation and speculation, respectively). In fact, these values are higher than those obtained by the Naïve Bayes and the SVM classifiers with the baseline configuration (see Tables 8 and 9). In the case of speculation, the result is even higher than the best performance obtained by Naïve Bayes (68.59% vs. 50.49% in terms of F_1 and 79.75% vs. 78.34% according to G-mean). This is due to the high precision yielded by the baseline. Almost the same occurs in terms of PCS and PCRS, where the baseline shows better performance than the two approaches with the basic set of attributes. However, as the final columns of Table 7 show, these results are subject to upgrading, for both negation (PCS = 23.07%; PCRS = 58.03%) and speculation (PCS = 13.86%; PCRS = 45.49%). This fact highlights that a simple configuration is not enough to detect the scope and that it is necessary to include more sophisticated features to successfully address the problem.

As explained in the Method section, Naïve Bayes is not the most suitable classifier to use for the task since its results are not satisfactory, and even lower than the baseline in some cases. For both negation and speculation, the best F_1 and PCS are achieved using the contextual configuration (see Table 8). However, the best PCRS (77.71% for negation, 64.30% for speculation) and G-mean (89.23% in negation, 78.34 in speculation) are obtained after adding syntactic information. This results from the fact that they are related to the recall. Conversely, F_1 as well as PCS are affected by the precision (i.e., a higher precision, higher F_1 or PCS). Therefore, in this case, contextual information seems to enhance the precision, whereas syntactic information improves the recall.

The classifier that best fits the data is SVM. The best results, as Table 9 shows, are obtained by adding syntactic

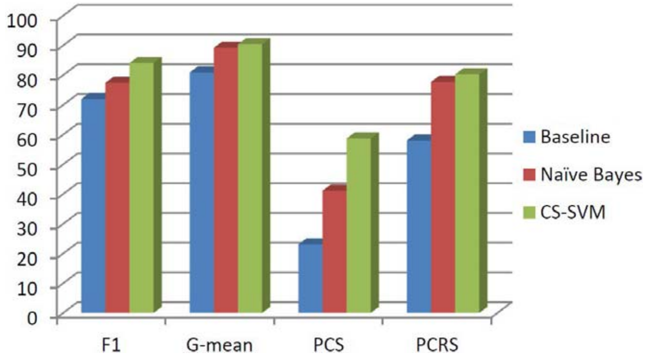


FIG. 5. Comparison of the results obtained by the different approaches in the negation scope detection task in terms of F_1, G-mean, PCS, and PCRS (%). [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

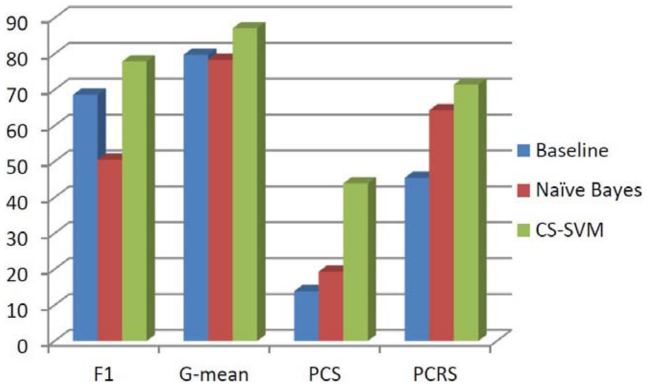


FIG. 6. Comparison of the results obtained by the different approaches in the speculation scope detection task in terms of F_1, G-mean, PCS, and PCRS (%). [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

information and applying cost-sensitive learning (CS-SVM) to solve the imbalanced data set problem. This algorithmic-level solution is effective in this case because the classes are highly imbalanced. However, although the improvement introduced by CS-SVM is substantial in many cases, it cannot be considered statistically significant, as revealed by the two-tailed sign test (in negation, $p = .56$, .55, .50, and .35 for F_1, G-mean, PCS, and PCRS, respectively; in speculation, $p = .54$ for F_1, $p = .56$ for G-mean, $p = .68$ for PCS and $p = .10$ in the case of PCRS). This configuration is favored by the stratified cross-validation whose results are slightly higher than those achieved in the random way. As the two-tailed sign test shows, the difference between them is not yet statistically significant ($p > .05$ in all cases).

In negation, the system yields an F_1 of 84.07% as well as G-mean, PCS, and PCRS values of 90.42%, 57.86%, and 79.13%, respectively. This means that the use of syntactic features (together with an algorithmic level solution to tackle the imbalanced data set problem) significantly improves the basic configuration by more than 40% in terms of F_1 and PCS, 30% according to G-mean, and the double in terms of PCSR. In addition, the configuration based on contextual features is also significantly enhanced, as shown by the two-tailed sign test ($p < .05$ in all cases). This improvement is higher in terms of percentage of correct scopes identified, where adding syntactic information exceeds it by almost 6%. Under this measure, there is also a significant difference if CS-SVM is compared with both the baseline ($p = 3.06^{E-17}$) and the Naïve Bayes classifier ($p = 2.82^{E-10}$) as Figure 5 shows. Derived from the figure, considerable differences can also be observed between CS-SVM and the other approaches in terms of PCRS and $F_{1.}$

In speculation, as mentioned before, the results are lower than those obtained in negation. In terms of F_1 (78.88%) and G-mean (87.14%), there is an improvement on the baseline (by roughly 10 percentage points in F_1 and 7% according to G-mean). This proportion is higher if we compare it to Naïve Bayes (almost 28% comparing F_1 value and 9% in G-mean). In terms of PCSR (71.43%) and, especially, in PCS (43.94%), the results could be improved. However, CS-SVM outperforms the baseline and the Naïve Bayes classifier by more than 24 percentage points in terms of PCS, a difference statistically significant ($p = 1.58^{E-12}$ compared to the baseline; $p = 2.46^{E-15}$ compared to Naïve Bayes). According to the PCRS measure, the CS-SVM classifier substantially outstrips the baseline results by more than 25% as well as obtaining about 7% more than the Naïve Bayes classifier. All these differences in performance are displayed graphically in Figure 6.

Interannotator agreement for negation and speculation (81.88% and 70.20% in F_1 measure, respectively) reveal the difficulty of the task. At the same time, the results stress that scope is an issue of the cue, the context, and the syntactic structure of the sentence taken together.

Finally, Table 10 shows the results of the whole system, i.e., using as cues those detected by the SVM classifier in the previous phase. These cues have been predicted applying the postprocessing step. To identify the scope, the CS-SVM classifier with contextual and dependency syntactic features was used since it is the configuration that yields the best result using the gold standard cues.

In general, the results are lower due to the errors that the classifier introduces in the cue detection and which are accumulated in the scope recognition phase. In negation, the system performance drops by between 4% and 10% depending on the measure (about 9% in F_1, 4% in G-mean, 7% in PCS, and 10% in PCRS). This difference is lower in speculation, where the results fall by 3% in terms of PCS and about 5% with regard to F_1, G-mean, and PCRS measures. It can be explained by the good performance achieved by the classifier in the speculation cue detection (F_1 values of 92.32% in the random way and 92.37% in the stratified one) which is comparable to those obtained by an annotator doing the same task. This suggests that when a cue is correctly predicted, its scope is also properly identified.

The results are promising and the system is portable. They are higher than the baseline results, especially in terms

TABLE 10. Results for detecting negation and speculation scopes with predicted cues: Averaged 10-fold cross-validation results for the **CS-SVM** classifier on the SFU Review corpus training data. Results are shown in terms of Precision, Recall, F_1, G-mean, PCS, and PCRS (%).

	Random						Stratified					
	Prec	Rec	F_1	G-M	PCS	PCRS	Prec	Rec	F_1	G-M	PCS	PCRS
Negation	72.09	76.72	74.33	86.77	51.33	69.58	72.06	76.98	**74.43**	**86.86**	**51.49**	**69.69**
Speculation	78.36	70.32	74.12	82.88	40.47	65.45	79.14	70.36	**74.49**	**82.94**	**40.99**	**65.77**

Note. Same notes as in Table 9 apply.

TABLE 11. Performance of negation scope detection of the proposed system and the approaches developed by Councill and Lapponi in terms of PCRS with gold standard cues and the predicted ones (%).

	Gold-standard cues	Predicted cues
Councill et al.	–	39.80
Lapponi et al.	67.85	48.53
Our system	80.26	69.69

TABLE 12. Errors in the cue detection phase.

	Negation	Speculation
False negative errors		
Incorrect classification of an MWC	99	–
Words annotated as cues in just a few instances	41	121
Words mostly annotated as the opposite type	38	29
Cues with low frequencies of occurrences	28	73
Unclassified	33	95
Total	**239**	**318**
False positive errors		
Words that are cues in most of the cases	570	446
Incorrect classification of an MWC	75	23
Words mostly annotated as the opposite type	27	37
Unclassified	28	8
Total	**700**	**514**

MWC = multiword cue.

of PCS, where the system outstrips it by about 28% both in negation and speculation. This is relevant since PCS is a scope-based measure and not a token-based measure such as F_1. In speculation, the performance (according to F_1 and G-mean) is even higher than those shown by the Naïve Bayes classifier, while in negation, this approach only exceeds it in terms of PCS.

Lastly, no significant differences were observed between randomly selecting and balancing the number of documents in each of the cross-validation folders.

Note that, as in the cue identification phase, the document-level partitioning of the data for cross-validation could have slightly deflated the results of the scope detection.

Comparison with previous works is not easy because they use different experimental settings, collections of documents, evaluation measures, etc. In addition, the results presented here cannot be directly contrasted with previous research since, to the best of our knowledge, there is no work related to recognizing negation and/or speculation using the SFU Review corpus. This is also a novel approach to detecting speculation in the review domain. However, there are some works that focus on automatically identifying the negation and its scope in this domain (Councill et al., 2010; Lapponi et al., 2012). Although these systems take different approaches and use different documents for training and testing (as explained in the Related Work section), which makes direct comparison not possible, this could give an indication as to how good the results detailed in this paper are in relation to others in the same task and domain.

As detailed in Table 11, Lapponi et al. obtained a PCRS value of 67.85% using the gold standard cues and 48.53% using the predicted ones. On their part, Councill et al. only specify the results by the whole system, which achieved 39.80% in terms of PCRS. The best configuration achieved in this paper yields 80.26% for the gold standard cues and 69.69% for the predicted ones. This highlights, once again, the difficulty of the task and shows that the results obtained by our system are in line with the results of other authors in the same task and domain.

Error Analysis

An analysis of the type of errors encountered in the SFU Review corpus system is detailed in this section. In the cue detection task, the analysis was done on the SVM approach (using the random cross-validation for speculation and the stratified one for negation, applying in this last case postprocessing), which performs best. The errors are summarized in Table 12 and are mainly due to the ambiguity that characterizes this type of document. In addition, many of them are related to the incorrect classification of MWCs.

Errors could be divided into two different categories: false-negative errors (FN) and false-positive ones (FP). In the first type of error, the system does not identify as cues words that are marked as such in the collection of documents. In negation, a total of 99 (41.4%) of them are the result of an incorrect classification of MWCs like *does n't* or *are not* where the system only annotates part of the cue (85 of them are corrected by the postprocessing algorithm). In 41 cases (17.15%) for negation and 121 (38.05%) for speculation, errors are words that appear annotated as cues in just a few instances in the corpus, so distinguishing the different usages from each other can sometimes be difficult, even for

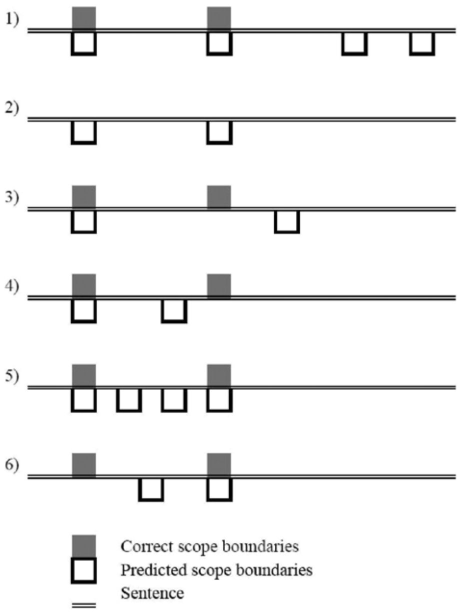


FIG. 7. Errors in the scope detection phase.

a human. Another type of error is related to cues that appear mainly annotated as the opposite type. Here, the classifier fails in 38 (15.89%) cases for negation and 29 (9.11%) for speculation. The last type of error is caused by cues with low frequencies of occurrence in the corpus. Examining more closely the distribution of these words, it can be seen that they appear only once and are due to annotation errors that arise out of spelling mistakes. Therefore, it is difficult for the algorithm to learn from examples. This error appears 28 times (11.71%) in negation and 73 times (22.95%) in speculation.

In the FP errors, the system recognizes as cues words that do not appear annotated as such in the corpus because the vast majority of these cases are due to the fact that the system indentifies as cues some words that appear in the corpus mostly classified as such (446 cases in speculation and 570 in negation). In contrast, 75 times (10.71%) in negation and 23 times (4.47%) in speculation, the system identifies only part of an MWC. In negation, all of these cases are corrected by the postprocessing algorithm. In speculation, this cannot be resolved by the postprocessing algorithm since almost all the MWCs consist of more than two words. Finally, another type of error is introduced when the classifier identifies a word as a negation/speculation cue when it has the opposite type, simply because they mostly appear as such in the corpus (i.e., the classifier tends to annotate them as the majority class).

In the scope detection, errors come from the CS-SVM approach (adding contextual and syntactic features and doing the cross-validation in a stratified way for both speculation and negation), which is the approach that achieves the best results. The most frequent errors are detailed in Figure 7 and described below where examples that compare the correct scope annotation for a cue (Gold Standard, henceforth GS) with the prediction made by the system (System Detection, hereafter SD) are listed:

1. The scope of the cue is a consecutive block of words. However, the system identifies not only the correct scope but also identifies other separated words as belonging to it. This is one of the most common mistakes made by the classifier, which occurs in 27.65% of negation and 23.35% speculation.

I suggest [**that if you are in doubt**], you seek assistance. (GS)
I suggest [**that if you are in doubt**], you [**seek assistance**]. (SD)

2. As mentioned in the Text Collection section, 5.44% of the total of negation cues and 4.62% of the total speculation cues do not have an associated scope. In this case, the cue belongs to this kind of keyword but the system incorrectly predicts some words as inside the scope of it. This represents 8.27% of the total errors in negation and 6.47% in speculation.

3. The beginning of the scope is correct, but the classifier fails in that it extends the scope beyond its correct ending. This mistake appears in 10.63% of the negation instances. In speculation, it constitutes 8.65% of the total errors.

No [**multitude of frilly thin spokes**] or cross-mesh design here. (GS)
No [**multitude of frilly thin spokes or cross-mesh design here**]. (SD)

4. This error is similar to the previous one. The beginning of the scope is correct, but the system incorrectly reduces the number of words in the scope to the right. In negation, this type of failure represents 28.6%, whereas in speculation it occurs 21.34% of the time.

The DVD-rom [**could have been either a lite-on**] or [**a Samsung**]. (GS)
The DVD-rom [**could have been either a lite-on**] or [**a**] Samsung. (SD)

The gold standard annotation does not normally include the full stop as inside the scope. However, there are some cases in which it is included (maybe due to annotation errors). This fact sometimes confuses the classifier so that its scope detection matches with the gold standard except that the system does not include the full stop when the annotation does.

5. Another type of error is introduced when the classifier correctly identified the beginning and the ending of the scope but it fails by omitting some words. It constitutes 11.84% of the total errors in negation and 6.97% in speculation.

The computer never [**recognized either cards**]. (GS)
The computer never [**recognized**] either [**cards**]. (SD)

6. In the last type of error, the end of the scope detected by the classifier is correct. However, it identifies the beginning of the scope after the correct position. This kind of

mistake hardly affects negation (it occurs in 0.67% of the cases). In speculation, this error represents 6.97% of the total.

And she ain't [**no rosellini**]. (GS)
And she ain't no [**rosellini**]. (SD)

A Case Study of Negation/Speculation for Sentiment Analysis

As proposed by authors like Councill et al. (2010), it could be useful to measure the practical impact of accurate negation/speculation detection to check whether it helps to improve the performance in sentiment analysis. Thus, an extrinsic evaluation is carried out with the aim of investigating whether correct annotation of negation/speculation improves the results of the Semantic Orientation CALculator (SO-CAL) system (Taboada, Voll, & Brooke, 2008; Taboada et al., 2011), using the approach described here as a recognizer for this kind of information, rather than the search heuristics that SO-CAL is currently using.

SO-CAL is a lexicon-based system that extracts sentiment from text. It uses dictionaries of words annotated with their semantic orientation (polarity and strength) to compute the sentiment polarity (positive or negative) of each document. The SO-CAL system incorporates negation and speculation. The negation approach consists of a backwards search to determine whether the lexical item is next to a negator (negators are those included in a predefined list). If the item is affected by the presence of negation, it is shifted by a fixed amount (3 or 4 depending on the item's POS) and multiplied by 1.5. Speculation, which is defined as irrealis, is dealt with in a crude way. The approach simply ignores the semantic orientation of any item in the scope of an irrealis marker (i.e., within the same clause). That includes statements with questions at the end, with modals, and conditionals.

The effect of the negation/speculation detection system on sentiment classification was evaluated in the SFU Review corpus, employing 10-fold cross-validation. As previously described, these results are compared to those obtained by using the search heuristics implemented in the SO-CAL system. A simple baseline model that involves not applying any negation/speculation resolution was also considered. This could show us more clearly the impact of introducing the treatment of this kind of information. Table 13 shows the results for all configurations.

As in the rest of the paper, a two-tailed sign test was used with the aim of assessing the statistical significance of differences in performance. For comparisons between the baseline and the two other configurations, the Paired Observation two-tailed test was employed. This statistical test is used when two of the same measurements are taken from the same subject, but under different experimental conditions, i.e., to compare the performance of the SO-CAL system before detecting negation and speculation to its performance after identifying this kind of information. In both cases, a significance level of $\alpha = 0.05$ was assumed.

In general, the results show that the SO-CAL system is biased towards positive polarity, with the F_1-score for positive reviews higher than it is for the negative ones. This difference is especially relevant in subcollections such as *Cookware*, where the number of positive expressions far exceeds the number of words that suggest a negative sentiment. However, these results are slightly balanced by introducing negation and speculation detection.

As expected, performance is improved by identifying this kind of information. In fact, all configurations that incorporate negation and speculation resolution outperform the baseline in terms of overall accuracy. Our proposed configuration, as shown in the final columns of Table 13, achieves the best performance, improving on the baseline by almost 10% and the search heuristics by about 5%. A two-tailed sign test reveals that there is not a statistically significant difference between the configuration proposed and the search heuristics approach (*p* = .259). However, as shown by the Paired-Observation two-tailed test, differences between

TABLE 13. Results of the SO-CAL sentiment classifier: Averaged 10-fold cross-validation results for SO-CAL including the neg/spe detector proposed, SO-CAL without including neg/spe treatment, and SO-CAL including neg/spe search heuristics on the SFU Review corpus training data. Results are shown in terms of F_1 for positive and negative reviews. Overall accuracy is also shown (%).

	Configuration								
	SO-CAL without neg/spe treatment			SO-CAL			SO-CAL with neg/spe detector integrated		
	Pos-F	Neg-F	**Accuracy**	Pos-F	Neg-F	**Accuracy**	Pos-F	Neg-F	**Accuracy**
Books	72.00	64.00	**68.00**	68.00	80.00	**74.00**	76.00	92.00	**84.00**
Cars	96.00	84.00	**90.00**	92.00	88.00	**90.00**	96.00	92.00	**94.00**
Computers	92.00	80.00	**86.00**	92.00	84.00	**88.00**	96.00	88.00	**92.00**
Cookware	92.00	32.00	**62.00**	84.00	48.00	**66.00**	92.00	56.00	**74.00**
Hotels	92.00	52.00	**72.00**	88.00	60.00	**74.00**	96.00	56.00	**76.00**
Movies	76.00	84.00	**80.00**	76.00	88.00	**82.00**	80.00	92.00	**86.00**
Music	84.00	68.00	**76.00**	80.00	72.00	**76.00**	84.00	76.00	**80.00**
Phones	88.00	52.00	**70.00**	84.00	72.00	**78.00**	84.00	72.00	**78.00**
Total	86.50	64.50	**75.50**	83.00	74.00	**78.50**	88.00	78.00	**83.00**

the method described and the baseline are significant ($p = .0019$), while those between the search heuristics implemented in the SO-CAL system and the baseline are not ($p = .19$).

In addition, for the positive reviews, only the proposed approach outstrips the configuration that does not include any treatment of negation/speculation (see second, fifth, and eighth rows of Table 13). These results can be explained by different factors. First, the detector presented in this paper benefits from a wider list of cues (search heuristics in SO-CAL include 14 different negation cues and 24 speculation cues, whereas the SFU corpus contains 69 and 129 different negation and speculation cues, respectively). This is crucial in speculation, where the number of occurrences of each cue is equally distributed across all documents. Second, the negation and speculation detection method proposed shows a good performance value, which suggests that when a cue is correctly predicted, its scope is also properly identified. In the approach based on search heuristics, a cue is identified only if it appears in the predefined list of cues, without taking into consideration whether it is actually acting as such. In addition, the scope is limited to certain parts of the sentence but it usually goes beyond the distance of words that the search heuristics method considers.

This illustrates that accurate detection of cues and scopes is of paramount importance to the sentiment detection task and, at the same time, it indicates that simplistic approaches to negation and speculation are insufficient for sentiment classification.

Finally, analyzing the cases in which the SO-CAL system does not detect the polarity of the reviews correctly (using as negation/speculation detector those described in this paper) helps to gain insight into the role of negation and speculation. Errors are mainly due to the fact that there many negative reviews which include a lot of positive expressions (in many cases with a high positive value) and in which the presence of negation/speculation is not very important. Therefore, it is difficult for the system to change the polarity of the review. The same occurs for positive reviews.

In addition, negation is not expressed by a cue in several cases. This means that the writer uses a positive statement followed by a conjunction and a negative one such as in *it's fine but I prefer another model*. This kind of expression is prevalent in the data.

Conclusion

This paper discusses a machine-learning system that automatically identifies negation and speculation cues and their scope in review texts. The novelty of this work lies in the fact that, to the best of our knowledge, this is the first system trained and tested on the SFU Review corpus annotated with negative and speculative information. In addition, this is the first attempt to detect speculation in the review domain. This is relevant since it could help to improve polarity classification such as that shown by Pang and Lee (2004).

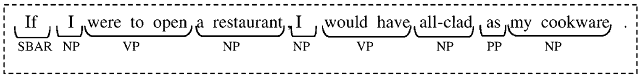


FIG. 8. Example of shallow parsing.

The results reported in the cue detection task (92.37% and 89.64% in terms of F_1 for speculation and negation, respectively) are encouraging. In the case of the speculation, the results are comparable with those obtained by a human annotator doing the same task. In the scope detection task, the results are promising in terms of F_1 (84.07% for negation and 78.88% for speculation), G-mean (90.42% and 87.14% for negation and speculation, respectively), and PCRS (80.26% in negation and 71.43% in speculation) but subject to improvement in terms of PCR (58.64% for negation and 43.94% for speculation).

The results show that, in line with comments by other authors, lexical information is enough to automatically identify the cues, whereas, to effectively determine the scope of a keyword, it is necessary to include syntactic features. An extrinsic evaluation is carried out with the aim of investigating whether correct annotation of negation/speculation improves the results of the SO-CAL system (Taboada et al., 2008, 2011), using the approach described here as a recognizer for this kind of information, rather than the search heuristics that SO-CAL is currently using. The results achieved demonstrate that accurate detection of cues and scopes is of vital importance to the sentiment detection task.

Future research includes the improvement of the scope detection results. Normally, the scope includes whole chunks, that is, sequences of words that form syntactic groups. Figure 8 shows an example where the cue is *if* and the scope consists of the phrases *were to open* and *a restaurant*. Shallow processing (chunking) applied in the postprocessing phase could help to correct the scope boundaries predicted by the classifier in the cases where they don't include complete syntactic group of words.

Acknowledgments

This work was supported in part by a grant to Maite Taboada from the Natural Sciences and Engineering Research Council of Canada (Discovery Grant 261104-2008). This work was partly funded by the Spanish Ministry of Education and Science (TIN2009-14057-C03-03 Project) and the Andalusian Ministry of Economy, Innovation and Science (TIC 07629 and TIC 07684 Projects).

References

Agarwal, S., & Yu, H. (2010). Detecting hedge cues and their scope in biomedical text with conditional random fields. Journal of Biomedical Informatics, 43(6), 953–961.

Akbani, R., Kwek, S., & Japkowicz, N. (2004). Applying Support Vector Machines to Imbalanced Data Sets (pp. 39–50). Machine Learning: ECML 2004. Pisa, Italy: Springer.

Apostolova, E., Tomuro, N., & Demner-Fushman, D. (2011). Automatic extraction of lexico-syntactic patterns for detection of negation and speculation scopes. Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: Short Papers (Volume 2, pp. 283–287).

Ballesteros Martínez, M. (2010). Mejora de la precisión para el análisis de dependencias usando maltparser para el castellano. Spain: Complutense University of Madrid.

Barua, S., Islam, M., Yao, X., & Murase, K. (2014). MWMOTE–majority weighted minority oversampling technique for imbalanced data set learning. IEEE Transactions on Knowledge and Data Engineering, 26(2), 405–425.

Benamara, F., Chardon, B., Mathieu, Y.Y., Popescu, V., & Asher, N. (2012). How do negation and modality impact opinions? Proceedings of the Workshop on Extra-Propositional Aspects of Meaning in Computational Linguistics (pp. 10–18).

Cao, P., Zaiane, O., & Zhao, D. (2014). A measure optimized cost-sensitive learning framework for imbalanced data classification. Biologically-Inspired Techniques for Knowledge Discovery and Data Mining, Advances in Data Mining and Database Management Book Series.

Chang, C., & Lin, C. (2011). LIBSVM: A library for support vector machines. ACM Transactions on Intelligent Systems and Technology (TIST), 2(3), 27.

Chapman, W.W., Bridewell, W., Hanbury, P., Cooper, G.F., & Buchanan, B.G. (2001). A simple algorithm for identifying negated findings and diseases in discharge summaries. Journal of Biomedical Informatics, 34, 301–310.

Choi, Y., & Cardie, C. (2008). Learning with compositional semantics as structural inference for subsentential sentiment analysis. Proceedings of the Conference on Empirical Methods in Natural Language Processing (pp. 793–801).

Councill, I.G., McDonald, R., & Velikovich, L. (2010). What's great and what's not: Learning to classify the scope of negation for improved sentiment analysis. Proceedings of the Workshop on Negation and Speculation in Natural Language Processing (pp. 51–59).

Cruz Díaz, N.P., Maña López, M.J., Vázquez, J.M., & Álvarez, V.P. (2012). A machine-learning approach to negation and speculation detection in clinical texts. Journal of the American Society for Information Science and Technology, 63(7), 1398–1410.

Dadvar, M., Hauff, C., & de Jong, F. (2011). Scope of negation detection in sentiment analysis. Proceedings of the Dutch-Belgian Information Retrieval Workshop (pp. 16–20).

Elkin, P.L., Brown, S.H., Bauer, B.A., Husser, C.S., Carruth, W., Bergstrom, L.R., & Wahner-Roedler, D.L. (2005). A controlled trial of automated classification of negation from clinical notes. BMC Medical Informatics and Decision Making, 5(1), 13.

Farkas, R., Vincze, V., Móra, G., Csirik, J., & Szarvas, G. (2010). The CoNLL-2010 shared task: Learning to detect hedges and their scope in natural language text. Proceedings of the Fourteenth Conference on Computational Natural Language Learning—Shared Task (pp. 1–12).

Gaifman, H. (1965). Dependency systems and phrase-structure systems. Information and Control, 8(3), 304–337.

Gelbukh, A., Torres, S., & Calvo, H. (2005). Transforming a constituency treebank into a dependency treebank. Procesamiento del Lenguaje Natural, 35(4), 145–152.

Gildea, D., & Jurafsky, D. (2002). Automatic labeling of semantic roles. Computational Linguistics, 28(3), 245–288.

He, H., & Garcia, E.A. (2009). Learning from imbalanced data. IEEE Transactions on Knowledge and Data Engineering, 21(9), 1263–1284.

He, H., & Ma, Y. (2013). Imbalanced Learning: Foundations, Algorithms, and Applications. Hoboken, NJ: John Wiley & Sons.

Hogenboom, A., van Iterson, P., Heerschop, B., Frasincar, F., & Kaymak, U. (2011). Determining negation scope and strength in sentiment analysis. 2011 IEEE International Conference on Systems, Man, and Cybernetics (SMC), 960, 2589–2594.

Horn, L.R. (1989). A natural history of negation. Chicago: University of Chicago Press.

Hsu, C., Chang, C., & Lin, C. (2003). A practical guide to support vector classification. Taiwan: Department of Computer Science and Information Engineering, National Taiwan University.

Huang, Y., & Lowe, H.J. (2007). A novel hybrid approach to automated negation detection in clinical radiology reports. Journal of the American Medical Informatics Association, 14(3), 304–311.

Jia, L., Yu, C., & Meng, W. (2009). The effect of negation on sentiment analysis and retrieval effectiveness. Proceedings of the 18th ACM Conference on Information and Knowledge Management (pp. 1827–1830).

Konstantinova, N., de Sousa, S., Cruz, N., Maña, M.J., Taboada, M., & Mitkov, R. (2012). A review corpus annotated for negation, speculation and their scope. Proceedings of the Eight International Conference on Language Resources and Evaluation (pp. 3190–3195).

Kumar, M., & Sheshadri, H. (2012). On the classification of imbalanced data sets. International Journal of Computer Applications, 44, 6280-8449.

Lapponi, E., Read, J., & Ovrelid, L. (2012). Representing and resolving negation for sentiment analysis. 2012 IEEE 12th International Conference on Data Mining Workshops (ICDMW) (pp. 687–692).

Liu, J., & Seneff, S. (2009). Review sentiment scoring via a parse-and-paraphrase paradigm. Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing (pp. 161–169).

Macdonald, C., & Ounis, I. (2006). The TREC Blogs06 collection: Creating and analysing a blog test collection. Department of Computer Science, University of Glasgow Tech Report TR-2006-224, 1, 3.1–4.1.

Martınez-Cámara, E., Martın-Valdivia, M., Molina-González, M., & Urena-López, L. (2013). Bilingual experiments on an opinion comparable corpus. WASSA 2013, 87.

Mitchell, K.J. (2004). Implementation and evaluation of a negation tagger in a pipeline-based system for information extraction from pathology reports. Proceedings of the Medinfo Conference (pp. 663–667).

Moilanen, K., & Pulman, S. (2007). Sentiment composition. Proceedings of the Recent Advances in Natural Language Processing International Conference (pp. 378–382).

Montoyo, A., Martínez-Barco, P., & Balahur, A. (2012). Subjectivity and sentiment analysis: An overview of the current state of the area and envisaged developments. Decision Support Systems, 53(4), 675–679.

Morante, R., & Blanco, E. (2012). * SEM 2012 shared task: Resolving the scope and focus of negation. Proceedings of the First Joint Conference on Lexical and Computational Semantics-Volume 1: Proceedings of the Main Conference and the Shared Task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (pp. 265–274).

Morante, R., & Daelemans, W. (2009a). Learning the scope of hedge cues in biomedical texts. Proceedings of the Workshop on Current Trends in Biomedical Natural Language Processing (pp. 28–36).

Morante, R., & Daelemans, W. (2009b). A metalearning approach to processing the scope of negation. Proceedings of the Thirteenth Conference on Computational Natural Language Learning (pp. 21–29).

Mutalik, P.G., Deshpande, A., & Nadkarni, P.M. (2001). Use of general-purpose negation detection to augment concept indexing of medical documents a quantitative study using the umls. Journal of the American Medical Informatics Association, 8(6), 598–609.

Nivre, J., Hall, J., & Nilsson, J. (2006). Maltparser: A data-driven parser-generator for dependency parsing. Proceedings of the Recent Advances in Natural Language Processing International Conference (pp. 2216–2219).

Özgür, A., & Radev, D.R. (2009). Detecting speculations and their scopes in scientific text. Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: 3, pp. 1398–1407.

Øvrelid, L., Velldal, E., & Oepen, S. (2010). Syntactic scope resolution in uncertainty analysis. Proceedings of the 23rd International Conference on Computational Linguistics (pp. 1379–1387).

Pang, B., & Lee, L. (2004). A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics (pp. 271–278).

Polanyi, L. (1986). The linguistic discourse model: Towards a formal theory of discourse structure. TR-6409. Cambridge: BBN Laboratories.

Polanyi, L., & van der Berg, M. (2011). Discourse structure and sentiment. 2011 IEEE 11th International Conference on Data Mining Workshops (ICDMW) (pp. 97–102).

Polanyi, L., & Zaenen, A. (2006). Contextual valence shifters. In J.G. Shanahan, Y. Qu, & J. Wiebe (Eds.), Computing attitude and affect in text: theory and applications (pp. 1–10). Dordrecht, Netherlands: Springer.

Rijsbergen, C.J.V. (1979). Information retrieval (2nd ed.). Newton, MA: Butterworth-Heinemann.

Rushdi Saleh, M., Martín-Valdivia, M.T., Montejo-Ráez, A., & Ureña-López, L. (2011). Experiments with SVM to classify opinions in different domains. Expert Systems with Applications, 38(12), 14799–14804.

Saurí, R. (2008). A factuality profiler for eventualities in text. (Ph.D. dissertation). Brandeis University, Waltham, MA.

Saurí, R., & Pustejovsky, J. (2009). FactBank: A corpus annotated with event factuality. Language Resources and Evaluation, 43(3), 227–268.

Sebastiani, F. (2002). Machine learning in automated text categorization. ACM Computing Surveys (CSUR), 34(1), 1–47.

Szarvas, G., Vincze, V., Farkas, R., & Csirik, J. (2008). The BioScope corpus: Annotation for negation, uncertainty and their scope in biomedical texts. Proceedings of the Workshop on Current Trends in Biomedical Natural Language Processing (pp. 38–45).

Taboada, M. (2008). SFU review corpus. Simon Fraser University. Retrieved from http://www.sfu.ca/~mtaboada/research/SFU_Review_Corpus.html

Taboada, M., Voll, K., & Brooke, J. (2008). Extracting Sentiment as a Function of Discourse Structure and Topicality. Vancouver, Canada: Simon Fraser University.

Taboada, M., Brooke, J., Tofiloski, M., Voll, K., & Stede, M. (2011). Lexicon-based methods for sentiment analysis. Computational Linguistics, 37(2), 267–307.

Velldal, E., Øvrelid, L., Read, J., & Oepen, S. (2012). Speculation and negation: Rules, rankers, and the role of syntax. Computational Linguistics, 38(2), 369–410.

Wiebe, J., Wilson, T., & Cardie, C. (2005). Annotating expressions of opinions and emotions in language. Language Resources and Evaluation, 39(2–3), 165–210.

Wiegand, M., Balahur, A., Roth, B., Klakow, D., & Montoyo, A. (2010). A survey on the role of negation in sentiment analysis. Proceedings of the Workshop on Negation and Speculation in Natural Language Processing (pp. 60–68).

Wilson, T., Wiebe, J., & Hoffmann, P. (2005). Recognizing contextual polarity in phrase-level sentiment analysis. Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing (pp. 347–354).

Witten, I.H., & Frank, E. (2005). Data mining: Practical machine learning tools and techniques. San Francisco, CA: Morgan Kaufmann.

Yessenalina, A., & Cardie, C. (2011). Compositional matrix-space models for sentiment analysis. Proceedings of the Conference on Empirical Methods in Natural Language Processing (pp. 172–182).

Zhu, Q., Li, J., Wang, H., & Zhou, G. (2010). A unified framework for scope learning via simplified shallow semantic parsing. Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing (pp. 714–724).

Zirn, C., Niepert, M., Stuckenschmidt, H., & Strube, M. (2011). Fine-grained sentiment analysis with structural features. Proceedings of the 5th International Joint Conference on Natural Language Processing (pp. 336–344).

Zou, B., Zhou, G., & Zhu, Q. (2013). Tree kernel-based negation and speculation scope detection with structured syntactic parse features. Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (pp. 968–976) Seattle, WA.

DOI: 10.1002/asi