Universidad de Huelva

Departamento de Tecnologías de la Información

Negation and speculation detection in medical and review texts

Detección de la negación y la especulación en textos médicos y de opinión

Memoria para optar al grado de doctora
presentada por:

Noa Patricia Cruz Díaz

Fecha de lectura: 10 de julio de 2014

Bajo la dirección del doctor:

Manuel Jesús Maña López

Huelva, 2014

Detección de la Negación y la Especulación en Textos Médicos y de Opinión

Tesis Doctoral

Presentada por

Noa Patricia Cruz Díaz

Programa de Doctorado:

Tecnologías Informáticas Avanzadas

E.T.S de Ingeniería

Universidad de Huelva

Huelva 2014

Noa Patricia Cruz Díaz

Tesis presentada para la obtención del título de

Doctor por la Universidad de Huelva

Dirigida por

Dr. Manuel J. Maña López

Grupo de Investigación:

Sistemas Inteligentes y Minería de Datos (TIC 198)

Programa de Doctorado:

Tecnologías Informáticas Avanzadas

E.T.S de Ingeniería

Universidad de Huelva

Huelva 2014

Negation and Speculation Detection in Medical and Review Texts

Ph.D. Thesis

Presented by

Noa Patricia Cruz Díaz

Doctoral program:

Tecnologías Informáticas Avanzadas

E.T.S de Ingeniería

Universidad de Huelva

Huelva 2014

Noa Patricia Cruz Díaz

A thesis submitted for the degree of

Doctor from the University of Huelva

Under the supervision of

Manuel J. Maña López. Ph.D

Research Group:

Sistemas Inteligentes y Minería de Datos (TIC 198)

Doctoral program:

Tecnologías Informáticas Avanzadas

E.T.S de Ingeniería

Universidad de Huelva

Huelva 2014

When you make the finding yourself – even if you're the last person on Earth to see the light – you'll never forget it.

Carl Sagan

Resumen

La detección de la negación y la especulación ha sido un área de investigación activa en los últimos años en la comunidad de Procesamiento del Lenguaje Natural, incluyendo algunas tareas competitivas en conferencias relevantes. De hecho, muchas aplicaciones se podrían beneficiar de la identificación precisa de este tipo de información (por ejemplo, *detección de interacciones*, *extracción de información*, *análisis de sentimientos*). Esta tesis tiene como objetivo contribuir a la investigación en curso sobre la negación y la especulación en la comunidad de la Tecnología del Lenguaje a través del desarrollo de sistemas de aprendizaje automático que determinen las palabras claves de negación y especulación así como resuelvan su ámbito lingüístico de aplicación. Entendemos por resolver el ámbito lingüístico, identificar a nivel de la frase los tokens que se ven afectados por las palabras claves. Se centra en los dos dominios en los que la negación y la especulación han recibido más atención: el biomédico y el de artículos de opinión. En el primero, el método propuesto mejora los resultados hasta la fecha para la sub-colección de documentos clínicos del corpus Bioscope. En el segundo, la novedad de la contribución radica en el hecho de que, hasta donde sabemos, éste es el primer sistema entrenado y evaluado en la colección de artículos de opinión Simon Fraser University anotado con información negativa y especulativa, al mismo tiempo, que supone el primer intento en detectar la especulación en este dominio. Además, y debido a los problemas de tokenización encontrados durante el pre-procesamiento de la colección de documentos BioScope y el escaso número de estudios en la bibliografía que aporten soluciones para este problema, la presente tesis describe este tema en profundidad proporcionando un análisis comprensivo así como lleva a cabo la evaluación de algunas herramientas de tokenización. Esta contribución supone el primer estudio de evaluación comparativo de tokenizadores en el ámbito biomédico, el cual podría ayudar a los desarrolladores de Procesamiento del Lenguaje Natural a elegir la mejor herramienta de tokenización a usar.

Abstract

Negation and speculation detection has been an active research area during the last years in the Natural Language Processing community, including some Shared Tasks in relevant conferences. In fact, it constitutes a challenge in which many applications can benefit from identifying this kind of information (e.g., *interaction detection, information extraction, sentiment analysis*). This thesis aims to contribute to the ongoing research on negation and speculation in the Language Technology community through the development of machine-learning systems which determine the speculation and negation cues and resolve their scope (i.e., identify at sentence level which tokens are affected by the cues). It is focused on the two domains in which negation and hedging have drawn more attention: the biomedical and the review domains. In the first one, the proposed method improves the results to date for the sub-collection of clinical documents of the BioScope corpus. In the second, the novelty of the contribution 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. At the same time, this is the first attempt to detect speculation in the review domain. Additionally, and due to the tokenization problems that were encountered during the pre-processing of the BioScope corpus and the small number of works in the bibliography which propose solutions for this problem, this thesis closely describes this issue and provide both a comprehensive overview analysis and evaluation of a set of tokenization tools. This means, the first comparative evaluation study of tokenizers in the biomedical domain which could help Natural Language Processing developers to choose the best tokenizer to use.

Contents

List of figures

List of tables

List of abbreviations

Abbreviation	Term
API	Application Programming Interface
BEP	Break-even point
CRC	Colorectal Cancer
CRF	Conditional Random Fields
ECG	Electrocardiogram
FAQ	Frequently Asked Questions
GS	Gold Standard
HPSG	Head-driven Phrase Structure Grammar
IE	Information Extraction
IR	Information Retrieval
KR	Knowledge Discovery
MAP	Mean Average Precision
MWC	Multi-word cue
NLP	Natural Language Processing
PCRS	Percentage of Correct Relaxed Scopes
PCS	Percentage of Scopes Correctly Classified
POS	Part-of-Speech
PPV	Positive Predictive Value
SD	System Detection
SFU	Simon Fraser University
SVM	Support Vector Machine
UMLS	Unified Medical Language System

Chapter 1

Introduction

1.1 Motivation

Negation and speculation are complex expressive linguistic phenomenas which have been extensively studied both in linguistic and philosophy (Saurí, 2008). They modify the meaning of the phrases in their scope. Negation denies or rejects statements transforming a positive sentence into a negative one, e.g., *Mildly hyperinflated lungs without focal opacity*. Speculation, also known as hedging, it is used to express that some fact is not known with certainty, e.g., *Atelectasis in the right mid zone is, however, possible*. These two phenomenas are interrelated (De Haan, 1997) and have similar characteristics in the text: they both have scope, so affect part of the text which is denoted by the presence of negation or speculation cue words.

The amount of negative and speculative information present in texts cannot be underestimated. Szarvas, Vincze, Farkas and Csirik (2008) report that 13.45% of the sentences in the abstracts section of the BioScope corpus and 13.76% of the sentences in the full papers section contain negations. In addition, they show that the percentage of sentences with hedge cues in the abstract and full papers section of the BioScope corpus are 17.70% and 19.44% respectively. In the review domain, this proportion is slightly higher. Konstantinova et al. (2012) show that 18% of the SFU Review corpus sentences contain negation cues and 22.7% of the sentences include speculation keywords. Therefore, the information that is inside the scope of any negation or speculation cue cannot be treated as factual. It should be discarded or presented separately with less confidence.

Nowadays, negation and speculation detection is an emergent task in Natural Language Processing (henceforth, NLP). In recent years, several challenges and shared tasks have

included the extraction of these language forms such as the *BioNLP'09 Shared Task 3* (Kim et al., 2009), the *CoNLL-2010 Shared Task* (Farkas, Vincze, Móra, Csirik, & Szarvas, 2010) or the *SEM 2012 Shared Task* (Morante & Blanco, 2012).

Detecting uncertain and negative assertions is relevant in a wide range of applications such as *information extraction* (henceforth, IE), *interaction detection*, *opinion mining*, *sentiment analysis*, *paraphrasing* and *recognising textual entailment* (Farkas et al., 2010; Konstantinova et al., 2012; Morante & Daelemans, 2009a; Morante & Daelemans, 2009b) . For all of these tasks it is crucial to know when a part of the text should have the opposite meaning (in the case of negation) or should be treated as subjective and non-factual (in the case of speculation). This part of the text is what is known as *scope*.

At first glance, negation and speculation might seem easy to deal with. The problem could be broken down into finding negative and hedge cues and determining their scope. However, it is much more problematic. Negation and speculation play a remarkable role towards understanding text and pose considerable challenges. They interact with many other phenomenas and they are used for so many different purposes that a deep analysis is needed (Blanco & Moldovan, 2011b).

This thesis is focused on the two domains in which negation and hedging have drawn more attention: the biomedical domain and the review domain. In the first one, negation and speculation detection can help in tasks like Protein-Protein interaction or Drug-Drug interaction. This particular area has been the focus of much current research, mainly due to the availability of the BioScope corpus (Szarvas et al., 2008); a collection of clinical documents, full papers and abstracts annotated for negation, speculation and their scope. In the review domain; opinion mining, sentiment analysis and polarity identification are examples of improvable tasks through the identification of negation and speculation. In all these tasks, distinguishing between objective and subjective facts is crucial and therefore negative and speculative information must be taken into account. Despite its importance and the interest of some authors to explore other areas apart from biomedical (Morante & Daelemans, 2012), the impact of negation and speculation detection in the review domain has not been sufficiently considered compared to the biomedical domain.

1.2 Negation and Speculation in Natural Language

A complexity in the natural language is the treatment of negation and speculation. In addition, it is a recurring theme in grammar. There are several reasons for this. Negation and speculation are not limited to the linguistic field but they have connections with many disciplines and domains, including philosophy, logic, mathematics or sociology.

As described in Horn and Kato (2000) , negation and hedging can be considered as a universal feature of the natural language, in the sense that all languages have a system to deny a statement or indicate uncertainty, in a way or another. Moreover, not only their existence seems to be universal, but the way in which each of the languages manifests, show they also move in a general direction.

However, despite this apparent uniformity, there is a wide variety of morphological and syntactic rules. Negation and speculation may be present in all units of the language (from the word to the discourse) and they also have important implications in morphology, phonetics, semantics, syntax or pragmatic levels.

The large number of publications and conferences held on this subject show their complexity and inherent relevance.

1.3 Biomedical domain

Medical practitioners are increasingly incorporating results and findings from clinical studies into their work. The availability of vast databases of scientific articles allows access to this material, although the huge volume also makes it difficult to locate relevant material. Furthermore, some hospitals have electronic records of their patients' medical background and many others are proceeding to digitize records. This enables physicians to carry out clinical studies which allow progress in evidence-based medicine. However, as in the case of access to scientific information, physicians need to have efficient tools to access this information. It is necessary to analyse the text in greater depth. This analysis should include negation and speculation detection because if not, automated indexing systems can suffer in terms of precision. For example, in Chapman's work (Chapman, Bridewell, Hanbury, Cooper, & Buchanan, 2001), when querying large medical free-text databases, the presence of

negations can yield numerous false-positive matches, because the medical staff is trained to include pertinent negatives in their reports. In a search for fracture in a certain radiology reports database, 95% to 99% of the returned reports would state *no signs of fracture* or words to that effect. Therefore, it is necessary to acknowledge whether words have been negated or not (Ballesteros, Francisco, Díaz, Herrera, & Gervás, 2012).

This is especially important in the biomedical domain, where negation and speculation are used extensively with the aim to express impressions, hypothesised explanations of experimental results or negative findings. An example is interaction extraction, where the aim is to mine text evidence for biological entities with certain relations between them. Here, an uncertain relation or the non-existence of a relation might be of some interest for an end-user so such information must not be confused with real textual evidence (Szarvas et al., 2008).

1.4 Review texts

Sentiment analysis is focused on the automatic detection and treatment of opinion in natural language applications. It is important for reasons such as *recommendation systems*, *affective computing* or *market research* (Lapponi, Read, & Ovrelid, 2012).

In this domain, hedges are linguistic means whereby the authors show that they cannot back their opinions with facts. Thus, speculations include certain modal constructions, along with other markers such as *indirect speech* (e.g., *according to certain researchers*). On the other hand, there are modal constructions which are not hedges, i.e., when expressing a factual possibility, without uncertainty on behalf of the speaker (e.g., *these insects may play a part in the reproduction of plants as well*) (Benamara, Chardon, Mathieu, Popescu, & Asher, 2012).

Negation is one of the most common linguistic means to change polarity (e.g., the polarity of the statement *Just a V-5 engine, spectacular* should be the opposite of its negation *Just a V-5 engine, nothing spectacular*). There are different types of negation such as *negative operators* (not, no more, without), *negative quantifiers* (nobody, nothing, never), *lexical negations* (lack, absence, deficiency), each of which has different effects on both the polarity and the strength of the negation. As authors like Benamara et al. (2012) discuss, 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. Hence, 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.

The literature on sentiment analysis and opinion mining (Councill, McDonald, & Velikovich, 2010; Dadvar, Hauff, & de Jong, 2011; Lapponi et al., 2012) 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.

1.5 Objectives and Contributions

The aim of this thesis is to contribute to the ongoing research on negation and speculation in the Language Technology community. In the medical domain, a system based on machine-learning techniques that identifies negation and speculation cues and their scope in clinical texts is proposed (Cruz, Maña, Vázquez, & Álvarez, 2012).

Additionally, and due to the tokenization problems encountered during the pre-processing of the BioScope corpus and the lack of guidance in this respect, this thesis closely describes this issue and provide both a comprehensive overview analysis and evaluation of tokenization tools. This means, the first comparative evaluation study of tokenizers in the biomedical domain which could help developers to choose the best tokenizer to use (Cruz & Maña, 2014).

In the sentiment analysis and opinion mining domains, and contrary to what happens in the biomedical field, there are no publicly available standard corpora of reasonable size annotated with negation and hedging. Therefore, as this thesis describes, the first step was the participation in the annotation process of the SFU Review corpus with negative and speculative keywords and their linguistic scope. It represents the first corpus annotated with this kind of information in the review domain. Next, using the corpora previously described as well as following the approach used in the biomedical domain, a system to automatically detect negation and hedge cues and their scope is presented (Cruz, Taboada & Mitkov, 2014).

1.6 Structure of the manuscript

An outline of the thesis is described below.

Chapter 2 begins with an introduction to the definition of negation and speculation from different perspectives, including a classification of the different types of each of them. After briefly motivating the importance of processing these language forms, this chapter presents the related work that inspired and motivated our work, both in the biomedical domain and in sentiment analysis.

Chapter 3 is dedicated to the tokenization problem in the biomedical domain with the aim of helping developers in the decision of chooising the best tokenizer to use. Therefore, this chapter provides an analysis of the problematic cases that the nature of the biomedical field introduces as well as a comprehensive comparative study of the available tools. Finally, it includes the evaluation of the 2 tokenizers that show better features and more accuracy and consistency in the previous study.

Chapter 4 is an in-depth description of the negation and speculation detection system for the clinical domain, explaining every step of the development process. It also presents the corpora used to build the system that accompanies it. Finally, this chapter describes how the system is evaluated and gives details about the experimentation, showing the results obtained and the discussion and error analysis around them.

Chapter 5 presents the developed system for the negation and hedging detection in review texts. It includes the description of the corpora used to train and test the system and the methodology followed. The corpora have been previously annotated for this task so their annotation process is also specified. It describes the evaluation process; the experiments performed as well as it details the system performance. A discussion and error analysis are also presented in this chapter.

Chapter 6 sums up the outcomes of the work done in this thesis and discuss the possibilities for future work.

This thesis is supplemented with a set of appendices whose content is the following:

- Appendix A. Description of the tokenization tools analysed in the tokenization problem in the biomedical domain.
- Appendix B. Set of sentences from the BioScope corpus used to test the tokenization tools.
- Appendix C. Output of each of the tokenization tools in the set of sentences from the BioScope corpus tested.
- Appendix D. Short description of the external libraries which have been used in the development of our systems and methodologies.
- Appendix E. List of publications related to this thesis, both in journals and conferences.

Chapter 2

Related work and background

2.1 Negation

2.1.1 Definition of negation

Negation is present in all languages and in its more obvious instance; it turns a proposition into its opposite. In a more sophisticated form, it is strongly expressive and includes euphemisms and irony. Unlike affirmative statements, negation is always marked by words (e.g., *not*, *without*), prefixes (e.g., *un-*, *in-*) or suffixes such as *–less* (Blanco & Moldovan, 2011a). In most cases, negation involves a cue and a negated syntagma which contains one or more words that are within the scope of negation (Ballesteros et al., 2012). For instance, in (1), *not* is the negation cue used to denote that the following concept (in this example, *expensive*) is negated.

(1) The chair is **not** *expensive* but comfortable.

However, negation is much more than a grammatical phenomenon present in all languages. It is *a linguistic, cognitive, and intellectual phenomenon* as Lawler (2010) affirms. Authors like Horn and Kato (2000) add that negation is *a central feature of language and cognition which interacts with all areas of grammar as well as with the philosophy of language*. In fact, negation in logic is well defined and syntactically simple (i.e., it is a unary operator which reverses the truth value) but in natural language, it is complex.

The study of negation from a philosophy perspective dates back to Aristotle. He defines the Law of Contradiction (a statement cannot be true and false at the same time) as well as the Law of Excluded Middle (a statement must be either true or false). After that, many studies

have been carried out in this regard, many of them collected in Seifert and Welte (1987). It should be highlighted the research conducted by Horn (1989) since it is currently considered a masterpiece. In this work, Horn lays out all the major topics concerning negation since Aristotle, and touches on negative polarity as well.

From a linguistic perspective, Tottie (1991) provides a quantitative analysis of negation, including a discussion about its linguistic variation. She discovers, for example, that there are twice as many negation words in speech as in writing (2.67 vs. 1.28, per 100 words). Valencia (1991) and Dowty (1994) study how negation influences reasoning while Hintikka (2002) support the argument that negation is a complex subject. At the same time, he explains that negation normally constitutes a barrier to anaphora as well as it interacts with quantifiers. In addition, he makes a distinction between contradictory and contrary negation. Van der Wouden (2002) defines the concept of negative context and deal with collocation, polarity and multiple negation. He argues that these topics are closely related since collocation is the general phenomenon of lexical items having a restricted distribution whereas polarity items are a specific class of such lexical items. He also adds that the same formal apparatus used to explain the behaviour of polarity items can be applied to others phenomena like some types of multiple negation. The aspects of polarity and multiple negation are also covered in The Cambridge Grammar of the English Language (Huddleston & Pullum, 2002) which includes a chapter tackling the negation problem. Recently, Morante and Sporleder (2012) summarise several aspects of negation and show that negative polarity and negation are different concepts, although interrelated. Basically, they explain that negation and polarity are related in the sense that negation can reverse the polarity of an expression. In this context, negative polarity items can be seen as expressions with a limited distribution, part of which includes negative sentences (e.g., *any* in the sentence *I didn't read any book*). Other works such as those presented by Laka (2013) explores negation from a syntactic point of view.

Finally, it is worth noting that negation is frequent in language. Indeed, as mentioned in Section 1.1., Szarvas et al. (2008) report that the number of negative sentences in the BioScope corpus is about 13% depending on the type of documents. Also in the biomedical domain, Nawaz, Thompson and Ananiadou (2010) explain that more than 3% of the biomedical events in 70 abstracts from the GENIA corpus (Kim, Ohta, Tateisi, & Tsujii, 2003) are negated. For their part, Councill et al. (2010) annotate a corpus of product reviews with

negation information, finding that 19% of the sentences contain negations. More recently, Konstantinova et al. (2012) show that 18% of the SFU Review corpus sentences include negative information. This proportion is higher in the ConanDoyle-neg corpus (Morante & Daelemans, 2012) where 22.49% of sentences are negative.

2.1.2 Types of negation

The major distinction can be made between **constituent** (or local) negation and **clausal** (or sentential) negation (Klima, 1964). A clausal negation negates an entire preposition (e.g., *he does not have money*) while a constituent negation is associated with some constituent or clause (e.g., *he has no money*). Although their effects can be similar or identical, the latest is less common grammatically.

Tottie (1991) presents the following comprehensive taxonomy of English clausal negation:

- Denials. They are the most common form and constitute unambiguous negations of a particular clause (e.g., *the audio system on this television is not very good, but the picture is amazing*).
- Rejections. They appear in expository text where a writer explicitly rejects a previous supposition (e.g., *given the poor reputation of the manufacturer, I expected to be disappointed with the device. This was not the case*).
- Imperatives. They involve an audience away from a particular action (e.g., *Do not neglect to order their delicious garlic bread.*).
- Questions. For instance, *why couldn't they include a decent speaker in this phone?*
- Supports and Repetitions. They express agreement and add emphasis or clarity. They involve multiple expressions of negation.

Tottie includes rejections and supports in intersentential negation (i.e., the language used in one sentence may explicitly negate a proposition or implication found in another sentence) while denials, imperatives, and questions are examples of sentential negation.

For its part, Payne (1997) defines different types of both clausal and constituent negation in any language. Clausal negation can be divided into the following categories:

- Lexical negation describes a situation in which the concept of negation is part and parcel of the lexical semantics of a particular verb.
- Morphological negation where the morphemes that express clausal negation are associated with the verb.
- Analytic negation in which the negative particles are normally associated with the main verb of the clause (e.g., *n't*, *not*, *never*).

The different types of constituent negation are described as follows:

- Derivational negation. Languages allow a stem to convert into its *opposite* by the use of some derivational morphology (i.e., suffixes and prefixes).
- Negative quantifiers. Many languages employ quantifiers that are either inherently negative (e.g., *none*) or are negated independently of clausal negation (e.g., *not many*).

Other authors determine different classes of negation in English. For example, Huddleston and Pullum (2002) identify the four contrasts for negation presented below:

- Verbal vs. Non-Verbal. In verbal, the negative particle is associated with the verb whereas in non-verbal the negation cue is related to a dependent of the verb.
- Clausal vs. Subclausal. A negation is clausal if it yields a negative clause. Otherwise, the negation is subclausal.
- Analytics vs. Synthetic. Negation in the analytic form is denoted by words whose only syntactic function is to mark negation. In synthetic, the words that mark negation have also other functions in the sentence.
- Ordinary vs. Metalinguistic. Ordinary indicates that something is not the case while metalinguistic does not dispute the truth but reformulating a statement.

Harabagiu, Hickl and Lacatusu (2006) distinguish two main classes of negation: overt (directly licensed) negations and indirectly licensed negations. The first one includes overt negative markers such as *n't*, negative quantifiers (e.g., *no*) and strong negative adverbs like *never*. The second consists of verbs or phrasal verbs (e.g., *fail*, *keep from*), prepositions such as *except*, weak quantifiers like *few* and traditional negative polarity items (e.g., *any more*).

2.1.3 Processing negation

From a NLP perspective, incorporate information about negation has been shown to be useful for a number of NLP, text-mining, and information retrieval (henceforth, IR) applications. For example, in the biomedical domain, Averbuch, Karson, Ben-Ami, Maimon and Rokach (2004) include negation detection in the task of context-sensitive medical information retrieval. The authors explain that the context of negation, a negative finding, is of special relevance because many of the most frequently described findings are those denied by the patient or subsequently *ruled out*. Hence, if negation is not taken into account in this task, many of the retrieved documents will be irrelevant. Denny, Miller, Waitman, Arrieta and Peterson (2009) identify QT[1] interval prolongation from electrocardiogram (ECG) impressions using a general purpose natural language processor. In this work, the authors apply a modified version of the NegEx algorithm (Chapman et al., 2001) to identify the negation. They assert that NLP with negation detection can extract concepts from ECG impressions with high accuracy. Most recently, Denny et al. (2012) investigate how NLP improves recognition of colorectal cancer (CRC) testing in an electronic medical record. As part of its NLP, they identify unified medical language system (UMLS) concepts found in each sentence along with information of its relevant context and information about whether or not the concept is negated. Also, an algorithm identifies negated phrases as well as common verbs and other modifiers that change the status of CRC-related testing (e.g., *refused*, *declined*). The results show that applying NLP to an electronic health record detects more CRC tests than either manual chart review or billing records review (i.e., queries based on the billing code) alone.

On the other hand, many authors have studied the role of negation in sentiment analysis task. Councill et al. (2010) define a system that can identify exactly the scope of negation in free text. Their system achieves an 80.0% F-score. The authors conclude that, as they expected, performance is improved dramatically by introducing negation scope detection. In a more recent work, Dadvar et al. (2011) investigate the problem of determining the polarity of sentiments in movie reviews when negation words, such as *not* and *hardly*, occur in the

[1] It is a measure of the time between the start of the Q wave and the end of the T wave in the heart's electrical cycle

sentences. The authors observe significant improvements in the classification of the documents after applying negation detection.

Negation recognition can also improve other tasks. For instance, Fiszman, Rindflesch and Kilicoglu (2006) report that one of the main causes of failure showed by their summarisation system is due to missed negation. Therefore, negative information should be taken into account. In the field of recognising textual entailment, i.e., recognise whether the meaning of one text fragment is entailed (can be inferred) from the other text, de Marneffe et al. (2006) show how negation influences some patterns of entailment. They focus on contexts which reverse monotonicity, such as *negations* and *quantifiers*. Snow, Vanderwende and Menezes (2006) describe a heuristic which allows them to predict false entailment. Also in this task, Androutsopoulos and Malakasiotis (2009) discuss the need to be careful with negations and other expressions that do not preserve truth values.

2.1.4 Negation detection in the biomedical domain

Studies on the problem of negation detection evolve from rule-based approaches to machine learning techniques. Among the first types of research, the one developed by Chapman et al. (2001) stands out. Their algorithm, NegEx, which is based on regular expressions, 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%[2]. 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) and Huang and Lowe (2007), who report that negated terms may be difficult to identify if negation cues are more than a few words away from them. To address this limitation in automatically detecting negations in clinical radiology reports, they propose a novel hybrid approach, combining regular expression with grammatical parsing. The

[2] The measures of effectiveness are explained in Sections 4.3.1 and 5.3.1 of this thesis

sensitivity of negation detection is 92.6%, the PPV is 98.6%, and the specificity is 99.8%. Apostolova, Tomuro and Demner-Fushman (2011) present a linguistically motivated rule-based system for the detection of negation scopes. The system rule set consists of lexico-syntactic patterns automatically extracted from the BioScope corpus which outperforms the baseline in all cases and exhibits results comparable to machine-learning systems.

However, most work in the field of negation detection is based on machine-learning approaches. Examples of detecting negated concepts in medical narrative using machine-learning techniques are the research by Averbuch et al. (2004) and Goldin and Chapman (2003).

It highlights the research conducted by Morante, Liekens and Daelemans (2008) which shows a high performance in all the sub-collections of the BioScope corpus. Their machine-learning system consists of two classifiers. The first one decides if the tokens in a sentence are negation cues. The second determines which words in the sentence are affected by the negation. They apply post-processing to increase the number of fully correct scopes. With this approach, the algorithm shows an F-score of 80.99% and 50.05% of scopes correctly identified. An improvement on this system is presented by the authors in 2009 (Morante & Daelemans, 2009b). In this case, they employ four classifiers instead of one to find the full scope of the negation cues. Three classifiers predict whether a token is the first token, the last, or neither in the scope sequence. A fourth classifier uses these predictions to determine the scope classes. To predict the cues, a list of 17 negation keywords extracted from the training data set is used. Instances with these negation cues are directly assigned to their class, so the classifiers only predict the class of the rest of the tokens. The set of documents employed for experimentation is wider (they use the whole BioScope corpus instead of just the abstracts as the previous system does). The third difference between these two approaches is that, in this case, a more refined set of attributes is used. 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%.

Another recent work is that developed by Agarwal and Yu (2010b). In this work, the authors detect negation 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. The authors select all negation sentences from the three sub-corpora and an equal number of non negation sentences randomly chosen. These new sub-corpora are divided into two groups; one is used for training and the other for testing. The best CRF-based model achieves an F-score of 98% and 95% on detecting negation cue phrases and their scope in clinical notes, and an F-score of 97% and 85% on determining negation cue phrases and their scope in biological literature. However, due to the fact that the corpus partitions and the evaluation measures are different, this system is not comparable with the approaches previously described.

An interesting approach to scope learning is those 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 cue. Evaluation on the BioScope corpus shows an F-score of 78.50% for abstracts, 57.22% for papers and 81.41% in the case of the clinical documents (using as cues those previously detected for the classifier). With the gold standard cues (those that appear annotated as such in the corpus), the results are notably higher. This means that this kind of systems together with an accurate cue classifier could be appropriated to tackle 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 which 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 et al. (2012) improves the results to date for the sub-collection 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 the way of selecting compatible attributes for different part-of-speech (POS) since features have

imbalanced efficiency for scope classification which is normally affected by the POS. Evaluation on the BioScope corpus reports an F-score of 76.90% in the case of the abstracts sub-collection, 61.19% for papers and 85.31% for clinical documents (using the gold standard cues).

2.1.5 Negation detection in sentiment analysis

In contrast to the biomedical domain, the impact of negation detection on sentiment analysis has not been sufficiently investigated, perhaps because standard corpora of reasonable size annotated with this kind of information has become available only recently. This motivated our participation in the annotation of a new corpus with negative and speculative information, i.e., the SFU review corpus (Konstantinova et al., 2012).

Many existing sentiment analysis approaches have relatively straightforward conceptualizations of the scope of negation keywords. For instance, Pang and Lee (2004) assume that the scope of a negation cue consists of the words between the negation keyword and the first punctuation mark following it. Kennedy and Inkpen (2006) introduce the concept of contextual valence shifters (i.e., negation, intensifier and diminisher). They experiment with taking as scope the remainder of the sentence as well as the first sentiment carrying word following the negation cue. Other approaches only consider specific types of words. For example, Hu and Liu (2004) suggest that the scope of negation is the adjectives that appear closely around the negation cue. They remark that the word distance between the negation keyword and the words in the scope should not exceed a threshold of about 5.

However, these solutions are not accurate enough. This is why research has been performing on integrating scope detection into sentiment analysis systems. Jia, Yu and Meng (2009) 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. This 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 associated scope. They 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 et al. (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). Using the same corpus, Lapponi et al. (2012) present a state-of-the-art system for negation detection. The heart of the system is the application of CRF models for sequence labelling which makes use of a rich of lexical and syntactic features, together with a fine-grained set of labels that capture the scopal behaviour of tokens. At the same time, they demonstrate that the choice of representation has a significant effect on the performance. Also in the review domain, Cruz et al. (2014) present a machine-learning system that automatically identifies negation cues and their scope in the SFU Review corpus (Konstantinova et al., 2012). The results obtained by this system are in line with the results of other authors in the same task and domain such as Councill et al. (2010) and Lapponi et al. (2012).

2.2 Speculation

2.2.1 Definition of speculation

The phenomenon of speculative language should be studied within the framework of modality since it involves, among others (i.e., subjectivity, evidentiality, uncertainty, committed belief, and factuality), the related concept of speculation.

Generally speaking, modality is what allows speakers to attach expressions of belief, attitude and obligation to statements. Morante and Sporleder (2012) present a great overview of the concept of modality from which it can be drawn that modality can be defined as a philosophical concept, as a subject of study in logic or a grammatical category. First, from a philosophical point of view, Von Fintel (2006) defines modality as *a category of linguistic meaning having to do with the expression of possibility and necessity*. He explains that there

are different types of modal mining (i.e., alethic, epistemic, deontic, bouletic, circumstantial and teleological) which can be conveyed by several types of expressions such as *conditionals*, *adjectives*, *nouns*, *adverbs, modal auxiliaries* and *semimodal verbs*. Next, within the modal logic framework, Kratzer (1981) analyses modality in terms of possible world semantics, where a proposition is identified with the set of possible worlds where it is true. She remarks that the interpretation of modals should consider a conversional background which implies that the meaning of modal expressions is context-dependent. Finally, from a grammatical perspective, Palmer (2001) defines modality as a valid cross-language grammatical category which is similar to aspect or tense since all three are categories of the clause as well as being concerned with the event or situation that is reported by the utterance. He considers that speculation falls within the category of epistemic modality because it is the means by which the speakers express judgement about the factual status of the proposition (e.g., *John may be in his office*).

The notion of speculation, also known as hedging, is first introduced by Lakoff (1972). He describes it as *words whose job is to make things fuzzier or less fuzzy*. Definitions are rare in the literature. Some examples are Zuck and Zuck (1986) who define hedging as *the process whereby the authors reduce the strength of a statement* and Markkanen and Schröder (1989) who consider it as *a manipulative, non-direct sentence strategy of saying less than one means*. Hyland (1995) refers to speculation as *the expression of tentativeness and possibility in language use*. He extensively studies the topic, focusing on scientific texts where statements are rarely made without subjective assessments of truth. In Hyland (1998), he explains that modality can be seen as *any linguistics means used to indicate either a lack of complete commitment to the truth value of any accompanying proposition or a desire not to express that commitment categorically*. In addition, he also argues that speculation is one part of epistemic modality because *it indicates an unwillingness to make an explicit and complete commitment to the truth of propositions*. Hyland establishes a categorization of hedge cues dividing them into lexical and non-lexical. The first one includes modal auxiliaries like *may* and epistemic modality: judgment verbs (e.g., *suggest*), evidential verbs (e.g., *appear*), deductive verbs (e.g., *conclude*), adjectives like *probable*, adverbs (e.g., *possibly*) and nouns like *suggestion*. Non-lexical features are referred to limiting experimental conditions, to model or theory, or to an admission of lack of knowledge. Others examples of authors who are studied hedging in the scientific domain are Light, Qiu and Srinivasan (2004) and Medlock and Briscoe (2007).

What does seem clear is that, as negation, speculation is a challenging phenomena from a computational point of view as well. Two main tasks have been addressed in the computational linguistic community, i.e., the detection of hedge cues as well as the resolution of the scope of these cues. For instance, in (2), the speculation cue is the token *could* while its associated scope is the syntagma *happen to him is an industrial accident*.

(2) The best thing that **could** *happen to him is an industrial accident*.

As the example shows, speculation cues are linguistic devices that reveal the author's attitude or opinion by presenting the information as uncertain or unreliable within the text (Verbeke et al., 2012). Hedge keywords can be expressed by different word classes as well as by multiword expressions (i.e., expressions that contain more than a word and whose meaning cannot be derived from the individual meanings of the words that constitute it) such as *cannot be excluded*. In addition, it becomes crucial to know, at sentence level, which words are affected by the cues.

Finally and similar to what happens with negation, speculative language is extensively used. Hyland (1996) reports one hedge in every 50 words of a corpus of research articles. Light et al. (2004) mention that 11% of the sentences in MEDLINE contain speculative language. Szarvas et al. (2008) explain that about 18% of the sentences in the abstract section and about 20% of sentences in the full papers sub-collection of the BioScope corpus correspond to speculation. In the review domain, Konstantinova et al. (2012) show that the percentage of speculative information in the SFU Review corpus is 22.7%.

2.2.2 Processing speculation

Some NLP applications, like IE, aim at extracting factual information from texts. As Prabhakaran, Rambow and Diab (2010) point out, *there is more to meaning than just propositional context*. They also argue that text cannot be seen as a repository of propositions about the world since language provides cues for the discourse participants to model cognitive state (i.e., beliefs, desires, and intentions).

Identifying speculative information is crucial in tasks such as sentiment analysis where, as Saurí and Pustejovsky (2009) explain, the same situation can be presented as a fact in the world, a mere possibility or a counterfact according to different sources. In fact, Pang and Lee (2004) show how subjectivity detection in the review domain helps to improve polarity classification. Wilson, Wiebe and Hoffmann (2005) also 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.

In the biomedical domain, Light et al. (2004) explore the use of speculative language in MEDLINE abstracts focusing on expressions of levels of belief (i.e., hypotheses, tentative conclusions, hedges, and speculations). They discuss how beneficial could be detect this kind of information in contexts like IE. For example, extracting tables of protein-protein interactions would benefit from knowing which interactions are speculative and which are definite. They add that, in the context of knowledge discovery (KR), current speculative statements about a topic of interest can be used as a seed for the automated knowledge discovery process. For its part, Medlock (2008) affirms that interactive bioinformation systems that take account of hedging can render a significantly more effective service to curators and researchers alike.

Recognising textual entailment is another task in which the speculation recognition is necessary. For instance, de Marneffe et al. (2006) capture simple patterns of modal reasoning, which illustrates the heuristic that possibility does not entail actuality. They map the text and the hypothesis according to the occurrence (or not) of predefined modality markers (i.e., possible, not possible, actual, not actual, necessary, and not necessary).

On the other hand, Baker et al. (2010) introduce modality identification in a machine translation application. They show how using a structure-based tagger to annotate English modalities on an English-Urdu training corpus improves the translation quality score for Urdu. They conclude that speculation is very important for a correct representation of events and likewise for translation.

Su, Huang and Chen (2010) explore how linguistically encoded information of evidentiality (i.e., linguistic representation of the nature of evidence for a statement) can contribute to the

prediction of trustworthiness, i.e., distinguish truth from lies, in NLP. Their experimental results using evidentials report improvements up to 14.85% over the baselines. This confirms that evidentiality is an important clue for trustworthiness detection.

In the classification of citations task, authors like Di Marco, Kroon and Mercer (2006) show that identifying the nature of the exact relationship between a citing and cited paper requires an understanding of the rhetorical relations within the argumentative context in which a citation is placed. To determine these relations automatically, the use of hedging to modify the affect of a scientific claim will be significant. They also explain that hedging is a relevant aspect of the rhetorical structure of citation contexts and that the pragmatics of hedges may help in determining the rhetorical purpose of citations.

Speculation detection is also beneficial in the field of identifying the text structure. For example, Grabar and Hamon (2009) study how the use of speculation markers within scientific writing may be useful for discovering whether these markers are regularly spread across biomedical articles and then for establishing the logical structure of articles. Exactly, they compute associations between article sections and speculation markers coming to the conclusion that speculation is governed by observable usage rules within scientific articles and can help their structuring.

2.2.3 Speculation detection in the biomedical domain

A fair amount of literature on hedging in scientific texts has been produced since the 1990s. For instance, Friedman, Alderson, Austin, Cimino and Johnson (1994) discuss uncertainty and hedging in radiology reports and their system assigns one of five levels of certainty (i.e., *no*, *low certainty*, *moderate*, *high* and *cannot evaluate*) to extracted findings.

However, speculative language from a NLP perspective has only been studied in the past few years. First approaches focus on detecting speculative sentences according to whether they contain speculation cues or not. Light et al. (2004) introduce the problem using a handcrafted list of hedge cues to identify speculative sentences in MEDLINE abstracts. They also experiment with automated methods proposing two different systems; one based on SVM, the other one based on substring matching. This latest approach marks as speculative those sentences that contain any of the following substrings: *suggest*, *potential*, likely, *may*, *at least*,

in part, *possible*, *potential*, *further investigation*, *unlikely*, *putative*, *insights*, *point toward*, *promise* and *propose*. Both the substring and the SVM systems perform well. The SVM classifier results are higher than those yielded by the substring matching method in terms of precision (84% vs. 55%). The opposite occurs in terms of recall where SVM obtains lower performance (39% compared with 79% for the substring matching method).

Medlock and Briscoe (2007) draw on this work and investigate automatic classification of speculative language using weakly supervised machine learning. They implement a simple probabilistic model for acquiring training data. This learner returns a labelled data set for each class, from which the probabilistic classifier is trained. The training corpus consists of 300,000 randomly selected sentences while they manually annotate six full-text papers from the functional genomics literature relating to Drosophila melanogaster (the fruit fly) to form the test corpus. They provide this dataset publicly available[3]. The system outstrips the baseline classifier described in Light et al. (2004) by 16% in terms of precision/recall break-even point (BEP). Error analysis shows that the model is unsuccessful in identifying assertive statements of knowledge paucity which are generally marked rather syntactically than lexically. In addition, the classifier also has difficulties in distinguishing between a speculative affirmation and one relating to a pattern of observed non-universal behaviour.

Medlock (2008) extends this work and experiments with additional features (POS tags, stems and bigrams). According to the results, they explain that adding POS and stems features to a bag-of-words input representation can slightly improve the accuracy. In addition, adding bigrams bring a statistically significant improvement over a bag-of-words representation. The best result outperforms the results previously obtained in Medlock and Briscoe (2007); 0.76 vs. 0.82 precision/recall BEP.

Szarvas (2008) follows Medlock and Briscoe in classifying sentences as being speculative or non-speculative. He extends their research by using a Maximum Entropy classifier which incorporates bigrams and trigrams as features, performing a re-ranking based feature selection procedure, and exploiting external dictionaries. In the experiments, he uses the dataset gathered by Medlock and Briscoe (2007) as a learning source at the same time that

[3] See *http://www.benmedlock.co.uk/hedgeclassif.html*

he makes available the BMC Bioinformatics data set[4] (by annotating four full text papers from the open access BMC Bioinformatics website) which is used for evaluation purposes. He investigates hedging in radiology reports as well. His best configuration (i.e., performing manual and automatic feature selection consecutively and using external dictionaries) achieves a precision/recall BEP performance of 85.29% and an F-score of 85.08% on the biomedical papers. He yields lower results on radiology reports (F-score of 82.07%). The error analysis indicates that more complex features like dependency structure and clausal phrase information could only help in allocating the scope of hedge cues detected in a sentence, not the detection of any itself.

Kilicoglu and Bergler (2008) apply a linguistically motivated approach to the same classification task by using knowledge from existing lexical resources and incorporating syntactic patterns. Additionally, hedge cues are weighted by automatically assigning an information gain measure and by assigning weights semi–automatically depending on their types and centrality to hedging. The system is evaluated on two different datasets: Drosophila data set from Medlock and Briscoe (2007) and the annotated BMC Bioinformatics papers from Szarvas (2008). In the first one, their approach achieves a competitive precision/recall BEP of 85% using the semi-automatic weighting scheme. On the BMC dataset, it yields a precision/recall BEP of 82%. The results confirm that selection of hedging devices affects the speculative strength of the sentence, which can be captured reasonably by weighting the hedge cues. Error analysis reveals that false positive errors are caused by the word sense ambiguity of speculation cues such as *could*, and by weak hedge cues like some adverbs (e.g., *usually*), normalizations (e.g., *implication*) and epistemic deductive verbs (e.g., *conclude*). False negative errors are due to the fact that the method does not address syntactic patterns and fails to identify certain derivational forms of epistemic words.

Shatkay, Pan, Rzhetsky and Wilbur (2008) introduce a novel task consisting of classifying sentence fragments from biomedical text along five dimensions. One of the dimensions is degree of certainty, according to which a statement could be assigned a value between 0 and 3, with 0 indicating no certainty and 3 indicating absolute certainty. Another dimension describes polarity which means the statement can appear negated or not. They annotate a

[4] See *http://www.inf.u-szeged.hu/~szarvas/homepage/hedge.html*

corpus of 10,000 sentences and sentence fragments selected from full-text articles from different biomedical journals. Using an SVM classifier, the results on level on certainty vary from 99% for level 3 to 46% for level 2, both in terms of F-score. Results on polarity classification are 95% F-score for the negative class and 1 F-score for the positive.

Ganter and Strube (2009) are the first authors in exploring a new domain. They develop a system for automatic detection of Wikipedia sentences that contain weasels. They adopt the Wikipedia's notion of *weasel words* (i.e., *words and phrases aimed at creating an impression that something specific and meaningful has been said, when it only a vague or ambiguous claim has been communicated*) since they are closely related to hedges and private states. The authors experiment with two different classifiers, one based on word frequency measures and another one based on syntactic patterns. Both approaches perform comparably well (around 70% precision/recall BEP) so word frequency and distance to the weasel tag is enough. The experiments also show that the syntactic patterns work better when using a broader notion of hedging tested on manual annotations.

In 2008, the availability of a resource which consists of clinical free-texts, biological texts from full papers and scientific abstract annotated for negation, speculation and their linguistic scope, i.e., the BioScope corpus[5] (Szarvas et al., 2008), facilitates the development of corpus-based statistical systems for negation/hedge detection. Since it was put publicly available, many works have been carried out using it as a training and evaluation source. In this sense, the task of resolving the cues and scope of speculation is first introduced in Morante and Daelemans (2009a). They port the system initially designed for negation detection (Morante & Daelemans, 2009b) described in Section 2.1.4 to speculation. In the first phase, hedge cues are identified by a set of classifiers, and in the second stage, another set of classifiers are employed to detect the scope of the speculation keyword. They show that the same scope-finding approach can be applied to both negation and hedging. 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%. Then, Özgür and Radev (2009) develop a supervised classifier for identifying speculation cues and a manually compiled list of lexico-syntactic rules for

[5] See *http://www.inf.u-szeged.hu/rgai/bioscope*

identifying their scopes. For the performance of the rule based system on identifying speculation scopes, they report 61.13% and 79.89% accuracy for the BioScope full papers and abstracts, respectively.

Using the same corpus, other authors have also taken into account speculation in their systems which, in most of the cases, have initially been designed for negation. For example, Agarwal and Yu (2010a) show an F-score of 88% and 86% in detecting speculation cue phrases and their scope in biological literature and 93% and 90% in clinical notes. However, as occurs in negation, their approach is not directly comparable due to the fact that they use different corpus partitions and evaluation measures. The system developed by Apostolova et al. (2011) reports an F-score of 75.57% for clinical documents, 78.99% for papers and 73.87% for abstracts in the scope recognition task. This means outperforming the baseline results as occurs in negation detection task. For its part, Cruz et al. (2012) get a performance value of 94.9% detecting the cues and 80.9% resolving the scope (with gold standard cues) in the clinical sub-collection, both in terms of F-score. Finally, the approach presented by Zou et al. (2013) 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 as cues those that appear annotated as such in the corpus).

This increased attention for speculation detection reflects in the fact that it has become a subtask of the BioNLP Shared Task in 2009 (Kim, Ohta, Pyysalo, Kano, & Tsujii, 2009), and the topic of the Shared Task at CoNLL 2010 (Farkas et al., 2010). The latter comprises two tasks: Task 1 is dedicated to detect uncertain sentences on two different domains, biological publications and Wikipedia articles. Task 2 aims to resolve the in-sentence uncertainty detection, i.e., automatically annotate the cue phrases and the left and right boundaries of their scope. In this case, the training and evaluation data consists of biological scientific texts.

In task 1, the best system for Wikipedia data is the developed by Georgescul (2010). It obtains an F-score of 60.2%. For biological documents, Tang, Wang, Wang, Yuan and Fan (2010) yield a performance of 86.4% in terms of F-score. Both approaches handle the task as a classical sentence classification problem and employ essentially a bag-of-words feature representation. In addition, neither of them derives features from syntactic parsing.

However, many authors tackle the task as a word-by-word token classification problem, i.e., they focus on the cue phrases and sought to classify every token if it is a part of a cue phrase, then a sentence is predicted as uncertain if it contains at least one recognised cue phrase. Examples are the approaches of Velldal, Øvrelid and Oepen (2010) and Vlachos and Craven (2010).

Task 2, for its part, it is implemented by all the authors as a two-stage-architecture where the speculation cues are first detected and then, the scope associated to these cues is predicted. The best result on hedge cue recognition (F-score of 81.3%) is obtained by Tang et al. (2010). Similarly to Morante and Daelemans (2009a), they set out to label words according to a BIO-scheme (i.e., determining whether the token is at the beginning, inside or outside of a hedge cue). They use a cascade subsystem in which a CRF model and a large margin-based model are trained. Then, another CRF model is trained using the result of the first predictions. For scope detection, the best F-score (57.3%) is yield by Morante, Van Asch and Daelemans (2010). They basically introduce some changes to the approach described in Morante and Daelemans (2009a): it uses one classifier per task instead of a metalearner combining three classifiers; information is added from the dependency tree instead of using shallow features only and a better treatment of multiword cues is carried out. Rei and Briscoe (2010) combine a set of manually compiled rules, a CRF classifier, and a sequence of post-processing steps on the same task, obtaining the second best result. Finally, Velldal et al. (2010) develop handcrafted rules based on syntactic information taken from dependency structures. With this approach, they achieve an F-score of 55.3%, the third best for the task.

As a follow-up of the CoNLL Shared Task and using the same corpora as training and evaluation sources, many systems have been implemented in the literature. Among them, it is worth highlighting the approach presented by Velldal et al. (2012). In the cue detection phase, they show a greatly simplified method to cue identification using a linear SVM classifier. This shall be accomplished by treating the set of cue words as a closed class. This means that the classifier only attempts to *disambiguate* known cue words, while ignoring any words not observed as cues in the training data. In the scope recognition phase, they employ a set of rules on syntactic features and n-gram features of surface forms and lexical information together with a machine learning system that selects subtrees in constituent

structures. The F-score reported by this system is 59.4% which outstrips the previous results in this task.

2.2.4 Speculation detection in sentiment analysis

As mentioned in Section 2.2.2, distinguishing between objective and subjective facts is crucial for sentiment analysis since speculation is a linguistic expression that tends to correlate with subjectivity (also known as private state). For instance, 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. However, although it has been proven that speculation has an effect on the opinion expression and it should be taken into account, there is, as far as we are aware, no work in detecting the speculation in the review domain. This is due to the fact that the annotation of a corpus with this kind of information, i.e., SFU Review corpus (Konstantinova et al., 2012), which would make it possible to tackle this problem efficiently, is recent. Using this corpus, Cruz et al. (2014) present the first attempt to detect speculation in the review domain. In addition, the results are promising both in cue and scope detection tasks.

2.3 Conclusions and chapter summary

This chapter is an overview of the concepts of negation and speculation and the major topics concerning them. Both are complex subjects which have been studied for a long time. Negation dates back to Aristotle and in its most trivial level, it reverses the truth value of a preposition. However, in more subtle examples it is strongly expressive and includes irony and euphemisms. Speculation, also known as hedging, can be defined within the framework of modality. It is first introduced by Lakoff (1972) and it is used by the speakers to present the information as uncertain or unreliable within the text.

In addition, this chapter has shown that negation and speculation detections have been an active research area during the last years in the NLP community, including some Shared Tasks in relevant conferences. In fact, they constitute a challenge in which many applications can benefit from identifying this kind of information (e.g., *recognising textual entailment*,

sentiment analysis, *IE*). Main tasks have been focused on determining the speculation and negation cues and the resolution of their scope (i.e., identify at sentence level which tokens are affected by the cues).

This thesis tackles negation and speculation treatment in computational linguistics in the two fields which have received more attention: biomedical and review. In the biomedical domain, many models for detecting keywords and resolving the scope have been proposed. However, much still remains to be done since scope detector performance is far from having reached the level of well established tasks such as *parsing*. In the review domain, although negation and speculation recognition can help to improve the effectiveness of sentiment analysis and opinion mining tasks, there is just a few works on detecting negative information. Besides, there is, as far as we are aware, no work in identifying speculation. Therefore, it is necessary to fill this gap through the development of systems which automatically identify both negation and speculation keywords and their scope.

Chapter 3

The tokenization problem in the biomedical domain

3.1 Motivation

Words and tokens in general, are the primary building blocks in almost all linguistic theories and language processing systems (Guo, 1997). Tokenization is the segmentation of text into these basic units for subsequent analysis (Webster & Kit, 1992). It is considered the first step in NLP together with the sentence splitting which obviously could affect the tokenization (Evang, Basile, Chrupała, & Bos, 2013). The result of this process is two types of tokens: one of them corresponding to units whose character structure is recognisable such as *punctuation* or *numbers*; the other being units which will need a morphological analysis (Grefenstette & Tapanainen, 1994). At first glance, all that seems to be involved is the recognition of spaces as word separators. However, in contrast to this perception, tokenization is a non-trivial problem (Jurafsky & James, 2000).

In the biomedical domain, tokenization is especially problematic due to several factors such as the atypical use of symbols and other irregularities, like technical terminology and new terms, tokenizer idiosyncrasies (e.g., *discarding or keeping dashes and hyphens*), a lack of guidance on how to adapt and extend existing tokenizers to new domains and inconsistencies between tokenizer algorithms (Barrett, 2012). An example of tokenization complexity in the biomedical domain could be biomedical substance names as they often contain special characters such as *slashes* or *brackets* (e.g., *N(2)-dimethylguanosine*) (Jiang & Zhai, 2007).

Tokenization is a fundamental processing step in many IR and IE tasks. All these tasks will be affected to a greater or lesser degree by decisions made in the process of tokenization. The way they are affected varies depending on the nature of the task and domain. IR tasks, such as *document retrieval*, are sensitive to small changes in the tokenization method (Trieschnigg, Kraaij, & de Jong, 2007). For example, Jiang and Zhai (2007) carry out a systematic evaluation of a set of tokenization heuristics on all the available TREC biomedical text collections for ad hoc document retrieval. Experiment results show that tokenization can significantly affect the retrieval accuracy where appropriate tokenization can improve the performance by up to 96%, in terms of mean average precision (MAP). In IE tasks such as *named entity recognition*, term variation is one of the most frequent causes of gene, protein and drug name recognition failures (Ananiadou, Kell, & Tsujii, 2006; Krauthammer & Nenadic, 2004). In another task like *negation detection*, Velldal et al. (2012) discuss the need of using an accurate tokenization since the effects of inaccurate tokenization might of course carry over to any downstream components using this information. In fact, they decide to adapt a cascaded finite-state tokenizer instead of using the Genia tagger because its tokenization rules are not always optimally adapted for the type of text used and therefore it introduces many errors.

It seems clear that choosing the right tokenizer is a non-trivial task that should be taken seriously since the biomedical domain poses additional challenges (He & Kayaalp, 2006) which if not resolved could mean the propagation of errors in successive NLP analysis pipelines. As a consequence, text-mining modules will inevitably suffer in terms of effectiveness (Tomanek, Wermter, & Hahn, 2007b). Therefore, this chapter aims to help developers choose the best tokenizer to use through a comprehensive overview study of tokenization tools in the biomedical domain (Cruz & Maña, 2014).

To the best of our knowledge, for the biomedical domain, there is only one work devoted to a systematic comparison of several tokenizers (He & Kayaalp, 2006). In this study, He and Kayaalp apply 13 tokenizers to 78 biomedical abstracts from MEDLINE, a corpus of biomedical literature compiled by the U.S. National Library of Medicine. They compare the outputs of these tokenizers showing that the results vary widely and only 3 of them produce identical output. Although the authors provide tokenizer characteristics, they do not specify which of them are important. Our contribution includes, not only the tools analysed by these authors but also those that they cannot test in their work for technical reasons as well as

several relevant tokenizers that appear more cited in literature; which have been obtained through screening Google Scholar and biomedical papers. Altogether, 21 tools have been considered for an in-depth analysis according to predefined criteria and 13 of them are tested on a set of sentences (tools which are not available or that also show too many errors in the test done by He and Kayaalp have been excluded for testing).

There are many annotated corpora publicly available to the community such as the *GENIA corpus*,(Kim, Ohta, Tateisi, & Tsujii, 2003; Ohta, Tateisi, & Kim, 2002; Tateisi, Yakushiji, Ohta, & Tsujii, 2005), PennBioIE (Kulick et al., 2004) or *GENETAG* (Tanabe, Xie, Thom, Matten, & Wilbur, 2005) as well as others not publicly available like the *JULIE corpus* (Tomanek, Wermter, & Hahn, 2007a), which have been used successfully by many groups to develop or compare NLP tools for the biomedical domain (Chen & Sharp, 2004; Clegg & Shepherd, 2007; Kang, van Mulligen, & Kors, 2011; Lease & Charniak, 2005; Rinaldi, Schneider, Kaljurand, Hess, & Romacker, 2006; Tomanek et al., 2007b). However, since each of these corpus consists of text extracted from a particular genre (i.e., paper abstracts) they do not cover all characteristics of biomedical text. That is the reason why a test corpus has been constituted, which gathers documents from different text types, as is the case of the BioScope corpus (Szarvas et al., 2008). It is a freely available resource consisting of abstracts and full scientific papers and clinical free text. A subset of this collection have been manually annotated with the aim to create the ground truth or gold standard tokenization, following the conventions, strategies and recommendations suggested in the literature. An evaluation of the two tools that show better features according the predefined criteria, and with more accuracy and consistency in the examples have been also carried out with the aim of discussing how well these tools are suitable for the biomedical field as well as aiding the decision making process of the developer on choosing the best tokenizer for this domain. Furthermore, this contribution is, as far as we are aware, the first comparative evaluation study on tokenizers in the biomedical domain (Cruz & Maña, 2014).

3.2 Problematic cases

Tokenization may seem simple if we assume that all it involves is the recognition of a space as a word separator (Ricardo & Berthier, 2011). However, a closer examination will make it clear that a blank space alone is not enough even for general English (Jurafsky & James,

2000). As discussed in Section 3.1, tokenization in biomedical literature is particularly difficult due to the fact that General English differs from biomedical text in vocabulary and grammar (Barrett, 2012). In addition, scientific information has a particular structure (Harris, 2002). For example, Campbell and Johnson (2001) carry out three experiments to evaluate the syntactic dissimilarities between medical discharge summaries and everyday English, showing significant differences in syntactic content and complexity. Another feature of the biomedical literature is related to terminology (Krauthammer & Nenadic, 2004), whose purpose is to collect the names of substances, qualities and processes employed in the biomedical domain both by practitioners and in the course of biomedical research. Specialized terminologies include SNOMED CT for clinical medicine, the Foundational Model of Anatomy for anatomical structures, the International Statistical Classification of Diseases and Related Health Problems for health disorders, the Gene Ontology™ for molecular biology and the Current Procedural Terminology for medical procedures and so on (Olivier Bodenreider and Burgun, 2004). Terminology is inconsistently spelled and may vary from typographical errors to lower case and capitalized medication names. Furthermore, biomedical texts could be ungrammatical as well as often including abbreviations and acronyms. Biomedical terms contain digits, capitalized letters within words, Latin and Greek letters, Roman digits, measurement units, lists and enumerations, tabular data, hyphens and other special symbols. Another difficulty is ambiguity, i.e., words and abbreviations that have different meanings (homonymy). For these reasons, the identification of terminology in the biomedical literature is one of the most challenging research topics in the last few years in NLP and biomedical communities, and tokenization plays an important role in handling them.

This section details all the biomedical domain difficulties, together with sentences extracted from the BioScope corpus, in which authors such as Velldal et al. (2012) find problematic cases where tokenizers fail. In addition, what it is considered to be the correct tokenization in each of these difficult cases is described as well. To choose the correct tokenization it has researched in depth into the literature and has followed the conventions, strategies and recommendations suggested by the authors. The potential complexities in the tokenization process can be divided into two major categories: those that apply across all domains and those that are more likely to be found in biomedical corpora, where there is a large amount of technical vocabulary (Clegg, 2008), as it is detailed below:

Common English complexities

- Hyphenated compound words

  For example:

  *Normal chest **x-ray**.*

  ***2-year 2-month** old female with pneumonia.*

  *The discrepancy score is the negative logarithm of a **p-value** of Fisher's exact test.*

  *Such **general-purpose** algorithms have also been developed to date.*

  Most human readers in common English would intuitively see these constructions as consisting of two words joined by a punctuation symbol. However, in the biomedical domain, there are cases, such as *hyphens*, which often concatenate entity names with other words (e.g., *IL-2-specific*) or even with other entity names (e.g., *CD43-DC*) forming an indivisible block. Therefore, all hyphenated compound words have been kept as one token.

- Words with letters and slashes

  Slashes usually indicate alternatives (e.g., *differentiation/activation*) or measurement units (e.g., *ng/ml*). In addition, they often separate two or more entity references like *IL-12/CD34*. Furthermore, they may also denote the knock-out status of a certain gene with respect to an organism (e.g., *flt3L-/-mice*) (Tomanek et al., 2007b). For example:

  *The maximal effect is observed at the IL-10 concentration of 20 **U/ml**.*

  *This may represent areas of atelectasis **and/or** pneumonia.*

  *These results indicate that within the **TCR/CD3** cue transduction pathway both PKC and calcineurin are required for the effective activation of the IKK complex and NF-kappaB in T lymphocytes.*

  *An upstream segment contains tandem dinucleotide repeats **(CT)19/(CA)16**.*

  There are two main strategies: Producing one token or producing two or more tokens by delimiting at slashes. Slashes generally separate different concepts so the word has been split into three tokens in all cases.

- Words with letters and apostrophes

  Apostrophes can indicate possessive (e.g., *years'*), words with single quotation (e.g., *'syntenic hits'*) and names (e.g., *O'Neill*). Examples of these might be the following:

  > *The false positive rate (FPR) of our predictor was estimated by the method of* ***D'Haeseleer*** *and Church 1855 and used to compare it to other prediction datasets.*
  >
  > *Small, scarred right kidney, below more than 2 standard deviations in size for* ***patient's*** *age.*

  There are a variety of tokenization strategies. For example, humans typically see them as single words while the influential Penn Tree Bank tokenization algorithm splits such cases into two tokens. Here, two cases have been differentiated: In the first one, when the word indicates a name (e.g., *D'Haeseleer*), it is counted as one token. In the second case, in words with single quotation or when they indicate a possessive (e.g., *patient's age*), the word has been separated from the apostrophes (e.g., *patient*$_{\wedge}$*'s*).

- Words with letters and brackets

  There are basically four types of brackets: parentheses, square brackets, braces and angle brackets. For instance:

  > *Of these, Diap1 has been most extensively characterized; it can block cell death caused by the ectopic expression of reaper, hid, and grim* ***(****reviewed in* ***[****26****])****.*

  When brackets are part of the name as occurs in chemical terminology (e.g., *(IL-1)-responsive kinase*), of course, these symbols should be tokens on their own. In all other cases, the brackets have been split from words.

- Abbreviations in capital letters and acronyms

  An abbreviation is a shortened form of a word or phrase. Usually, but not always, it consists of a letter or group of letters taken from the word or phrase. It must be taken into account in any tokenization process. An example of this may be the one shown below:

*Mutants in Toll cueing pathway were obtained from **Dr.** S. Govind: cactE8,cactIIIG, and cactD13 mutations in the cact gene on Chromosome II.*

An acronym is an abbreviation formed from the initial components in a phrase or a word. These components may be individual letters (as in *SARS; severe acute respiratory syndrome*) or parts of words (as in *Ameslan; American Sign Language*).

Abbreviations and acronyms are commonly used in biomedical literature. For example, in the medical domain, writing favors brevity because time pressures often prevent medical specialists from describing clinical findings fully and abbreviations are a convenient way to shorten the sentences (Grange & Bloom, 2000).

Abbreviations and acronyms mainly refer to names, but abbreviations of adjectival expressions are often found in the biomedical domain (e.g., *CD8+* is an abbreviation of *CD8-positive*). Therefore, they have been considered as a single word.

- Words with letters and periods

  Words with a period at the end usually indicate end of sentence. However, they may merely be abbreviations, such as *i.e.* and *e.g.,* as shown in the following example:

  *Two stop codons of an iORF (**i.e.** the inframe and C-terminal stops) can be any combination of canonical stop codons (TAA, TAG, TGA).*

  When they indicate the end of a sentence, periods should be split. Otherwise, they have been kept together with neighboring words.

- Words with letters and numbers

  For example:

  *Selenocysteine and pyrrolysine are the **21st** and **22nd** amino acids, which are genetically encoded by stop codons.*

  Such words have been kept as a whole unit.

- Words with numbers and one type of punctuation

  Some simple examples for numbers are: large numbers (e.g., *390,926*), fractions (e.g., *1/2*), percentages (e.g., *50%*), decimals (e.g., *0.001*) and ranges (e.g., *2-5*). These punctuation marks are: comma, forward slash, percent, period and en dash. Good illustrations extracted from the BioScope corpus are the following:

  *A total of **26,003** iORF satisfied the above criteria.*

  *E-selectin is induced within **1-2** h, peaks at **4-6** h, and gradually returns to basal level by 24 h.*

  They have been considered as a whole.

- Numeration

  It is regarded as the act or process of counting or numbering. For instance:

  ***1.** Bioactivation of sulphamethoxazole (SMX) to chemically-reactive metabolites and subsequent protein conjugation is thought to be involved in SMX hypersensitivity.*

  They have been kept as one word.

- A hypertext markup symbol

  Some of the frequently observed hypertext markup symbols are *<* and *"* (for the double quotation mark). For instance:

  *Bcd mRNA transcripts of **<** or = 2.6 kb were selectively expressed in PBL and testis of healthy individuals.*

  These symbols have been taken as a whole unit.

- A URL

  An example would be the following:

  *Names of all available Trace Databases were taken from a list of databases at **http://www.ncbi.nlm.nih.gov/blast/mmtrace.shtml***

  URL has been considered as a whole.

Biomedical English complexities

- A DNA sequence

  For example:

  > *Footprinting analysis revealed that the identical sequence* ***CCGAAACTGAAAAGG****, designated E6, was protected by nuclear extracts from B cells, T cells, or HeLa cells.*

  DNA sequence must be treated as a unit.

- Temporal expressions

  For example:

  > *This was last documented on the Nuclear Cystogram dated* ***1/2/01****.*

  Temporal expressions such as *dates* have been treated as single words.

- Chemical substances

  They include several symbols which may (or may not) denote word token boundary symbols such as *parentheses*, *hyphens* and *slashes* (Tomanek et al., 2007b). Furthermore, chemical substances basically comprehend gene symbols, drug names and protein names, each of which has certain characteristics as described below.

  <u>Gene symbols</u>

  The names can indeed be divided into the following three categories (Proux, Rechenmann, Julliard, Pillet, & Jacq, 1998):

  - Names including special characters, i.e., upper cases, hyphen, digit, slash or brackets. For example, *Lam-B1* or *M(2)201*.
  - Names in lower case and belonging to the general English language. For instance, *vamp* or *zip*.
  - Names using lower case letters only without belonging to the language such as *zhr* or *sth*.

Drug names

In general, most drug names include particular letters from the chemical formula (e.g., *Tylenol*, which are generated from *n-aceryl-para-aminophenol*), generic names such as *Thalomid*, Latin or Greek terminology, parts or abbreviations of the company's name (e.g., *Baycol*, (*Bayer+colesterol*)), low-frequency letters of the alphabet such as *x* or *y* (e.g., *x-trozine*) as well as acronyms like *Tigan* (that means this is good against nausea) (Gantner, Schweiger, & Schlander, 2002).

Protein name

Protein names can also be partitioned into three categories from their structure (Fukuda, Tsunoda, Tamura, & Takagi, 1998):

- Single words in upper case, numerical figures, and non-alphabetical letters which are mostly derived from gene name (e.g., *p53*).
- Compound words with upper case letters, numerical letters, and non-alphabetical letters. (e.g., *(IL-1)-responsive kinase*).
- Single word with only lower case letters (e.g., *insulin*).

An example which appears in the BioScope corpus is the following:

> *We found **IL-2Ralpha** expression to be increased in BAL cells from involved sites of active pulmonary tuberculosis.*

The use of chemical and biological names with embedded punctuation is a particular source of ambiguity, although in this case it is clear that they must be considered as a whole.

3.3 Comparative study of tools

3.3.1 Corpus annotation

The document collection used in this study is a subset of the BioScope corpus, consisting of texts taken from 4 different sources and 3 different types so that it captures the heterogeneity of language used in the biomedical domain. In total, more than 20,000 sentences grouped into the following categories:

- Clinical documents: This part represents the major part of the corpus at document level. It was used for the clinical coding challenge (Pestian et al., 2007) organised by the Computational Medicine Center in Cincinnati, Ohio, in 2007.
- Full articles: Five articles from FlyBase and four articles from the open access BMC Bioinformatics Web site.
- Abstracts: 1,273 scientific abstracts obtained from the Genia corpus (Collier et al., 1999). These types of documents are the main targets for various text-mining applications, such as *protein interaction mining*, because of their public accessibility.

These three sub-collections differ in many aspects. Clinical documents are characterised for consisting of short sentences, written in a medical language that often includes lexical and grammatical errors. In paper and abstract sub-collections, sentence length is much longer than in clinical data and the style of the texts is also more literary, therefore allowing for a greater degree of linguistic richness.
Table 3.1 summarises the characteristics of each sub-collection.

	Clinical	**Full papers**	**Abstracts**
#Documents	1,954	9	1,273
#Sentences	6,383	2,670	11,871
#Words	41,985	60,935	282,243
Av. length sentences (#words)	7.73	26.24	26.43

Abbreviation is as follows: *'Av.'*: average.

Table 3.1: Characteristics of the BioScope corpus

To carry out the evaluation of the two systems that show better features, accuracy and consistency in the selection phase, 10% of the total of sentences of papers and clinical sub-collections of the BioScope corpus has been randomly selected to create the ground truth or gold standard tokenization, following the guidelines described in Section 3.2. Due to the size of the sub-collection of abstracts, in this case, 5% of the total of sentences have been taken. This means: 638 sentences for clinical, 267 sentences for papers and 594 sentences in the case of abstracts.

A second human annotator annotates 20% of these sentences from the original collection, selected randomly. The annotation has been done according to the guidelines used by the first annotator. During the annotation process, annotators are not allowed to communicate

with each other. Inter-annotator agreement using Kappa (Cohen, 1960) has been measured, treating the first annotator as the gold standard. The agreement is considered quite high for the three sub-collections (Landis & Koch, 1977). For clinical texts, a Kappa value of 0.972 has been obtained. The discrepancies between annotators occur in words with letters and apostrophes. The guidelines described in Section 3.2 specify that in this case it should *separate the word from the apostrophes* but it do not specify in how many tokens. Thereby, it depends on the interpretation of the annotator. The first annotator separates the words with apostrophe into two tokens (e.g., *patient␣'s*) whereas the second annotator separates them into three ones (e.g., *patient␣'␣s*). To avoid this problem, the guideline has been refined adding an example of annotation. For papers, a Kappa value of 1 has been achieved so the annotators agree in all cases. Finally, in the case of the abstracts sub-collection, the Kappa value is 0.99. The second annotator fails annotating "*controls*," as "*␣controls,␣*" since the correct annotation is "*␣controls␣'␣*". The annotators agree on the remaining cases. Based on these results, we can be confident that the corpus is annotated correctly, and that the annotation is reproducible.

3.3.2 Tool selection phase

There is a variety of available tools which can convert an input stream of characters into a stream of words or tokens (i.e., tokenize a text). Although this work is focused on the biomedical domain, it has been taken into account not only tools designed for the biomedical area but also general purpose tools, in order to test if the tokenization process is explicitly related to the domain or the genre of the texts which are processed as some authors affirmed (Habert et al., 1998).

The criteria used to evaluate the selected tools consist of a checklist of 20 features developed according to the quality characteristics proposed by the ISO 9126-1 standard (ISO, 2001). These features can be classified as follow:

- Technical criteria which assess the system properties in general.
- Functional criteria. The aspect of functionality concerning the presence or absence of functions which are relevant for the tokenization. Roughly speaking, functionality concerns the relation tool–task.
- Usability. In contrast to functionality, usability takes user aspects into consideration by evaluating the effort needed for use; i.e., it concerns the relation tool–user.

Table 3.2 describes the set of criteria used in each of these categories.

With the aim of creating a list of potentially useful tools for tokenizing texts, a first selection of the tokenizers has been made from the work of He and Kayaalp (2006) where a representative number of tools that can be used for tokenization are analysed. This list is completed by adding: 1) some of the tools that He and Kayaalp do not test in their work for several technical reasons and 2) tokenizers which have been found more cited in literature through screening Google Scholar and biomedical papers, both designed for biomedical domain as well as for general purpose.

Category	Criteria
Technical	Year of publication
	Date of the last version
	Number of references in Google Scholar
	Type of installation: stand-alone, application programming interface (API)
	Availability of source code
	Supported operating systems
	Programming language
	Dependence on external software or libraries
	License of the tool
Functional	Part-of-Speech tagger associated
	Domain in which the tokenizer was trained
	Ability of reconstructing the text into its original format
	Possibility to train the tokenizer with other collections of documents
	Possibility to integrate in an application (i.e., API)
Usability	Learning curve
	Ease of use
	Ease of installation
	Availability and quality of the documentation
	Existence of support (mailing list, forum, FAQ, wiki)
	Quick start guide

Table 3.2: Criteria used for the evaluation of tokenization tools

A total of 21 tools are analysed in a first phase according to the predefined criteria previously detailed. Only published, available and constantly maintained tools have been considered for this analysis. Secondly and following the discussion and tests conducted by He and Kayaalp, those tokenizers that introduce a considerable number of errors in the experiment carried out by them (Brill's POS tagger, Dan Melamed's tokenizer, Gump tokenizer, LT TTT, Mallet tokenizer, MXPOST, Specialist NLP, UIUC word splitter) have been discarded. In a third phase, 13 of the initial 21 tools have been tested using a set of 28 sentences from the BioScope

corpus. These examples contain all the problematic cases that could be found in the biomedical domain which are detailed in Section 3.2.

Finally, to help developers choose the best tokenizer to use, an evaluation of the two tools that show more consistency and accuracy in the previous phase have been done. The evaluation is also carried out on a subset of the BioScope corpus. As mentioned in Section 3.3.1, to do this, a percentage of sentences (between 5% and 10%) of each sub-collection have been randomly selected and manually tokenized to create the ground truth or gold standard. Both tokenizers are tested and their accuracy is measured in terms of number of tokens that match with the gold standard. Errors or cases in which each tokenizer does not detect the tokens correctly have been also analysed.

3.3.3 Tool description

The tools analysed, based on the set of predefined criteria, are the following: *Brill's POS tagger*, *Dan Melamed's tokenizer*, *English Resource Grammar*, *Freeling*, *Genia tagger*, *Gate Unicode tokenizer*, *Gump tokenizer, JULIE LAB tokenizer*, *LingPipe*, *LT TTT*, *Mallet tokenizer*, *McClosky-Charniak parser*, *MedPost*, *MXPOST tagger*, *NLTK tokenizer*, *OpenNLP tokenizer*, *Penn Bio tokenizer*, *Stanford POS tagger*, *Specialist NLP*, *UIUC word splitter* and *Xerox tokenizer*. Table 3.3 details all these tokenizers showing their references and websites.

Some of the listed tools are available via web, while others have to be installed locally and could be accessed via command-line or graphical interface. It has been attempted to use all the tools with the aim of analysing their functionality and user interface as well as evaluating the easiness of installation and effort required to learn to use the tool.

This section describes each tool in detail including its reference, abstract, platform and ease of use, strengths, pitfalls, purpose, rules for doing the tokenization and so on. Tools are presented in alphabetical order. Table 3.4 summarises the main technical criteria of each tool. Table 3.5 details the functional ones while the usability criteria are shown in Table 3.6.

The complete description of each tool has been included in the Appendix A.

Tool	References	Website
Brill's POS tagger	(Brill, 1992)	http://www.umiacs.umd.edu/~jimmylin/resources.html
Dan Melamed's tokenizer (DMT)	-	http://www.cs.nyu.edu/~melamed/genproc.html
English Resource Grammar (ERG)	(Copestake & Flickinger, 2000; Flickinger, 2000)	http://www.delph-in.net/erg/
Freeling	(Carreras, Chao, Padró, & Padró, 2004; Padró & Stanilovsky, 2012)	http://nlp.lsi.upc.edu/freeling/
Genia tagger	(Kulick et al., 2004; Tsuruoka et al., 2005; Tsuruoka & Tsujii, 2005)	http://www.nactem.ac.uk/tsujii/GENIA/tagger/
Gate Unicode tokenizer (GUT)	(Cunningham, Maynard, Bontcheva, & Tablan, 2002)	http://gate.ac.uk/sale/tao/splitch6.html#sec:annie:tokeniser
Gump tokenizer	-	http://www.mozart-oz.org/mogul/doc/lager/gump-tokenizer/
JULIE LAB tokenizer (JLT)	(Tomanek et al., 2007b)	http://www.julielab.de/Resources/NLP+Tools.html
LingPipe	(Carpenter & Baldwin, 2011)	http://alias-i.com/lingpipe/
LT TTT	(Grover, Matheson, Mikheev, & Moens, 2000)	http://www.ltg.ed.ac.uk/software/lt-ttt2
Mallet tokenizer	(McCallum, 2002)	http://mallet.cs.umass.edu/index.php
McClosky-Charniak parser (MCP)	(McClosky & Charniak, 2008) (McClosky & Adviser-Charniak, 2010)	http://nlp.stanford.edu/~mcclosky/biomedical.html
MedPost	(Smith, Rindflesch, & Wilbur, 2004)	ftp://ftp.ncbi.nlm.nih.gov/pub/lsmith/MedPost/medpost.tar.gz
MXPOST tagger	(Ratnaparkhi, 1996)	http://www.inf.ed.ac.uk/resources/nlp/local_doc/MXPOST.html
NLTK tokenizer	(Bird, Klein, & Loper, 2009)	http://nltk.org/
OpenNLP tokenizer	-	http://opennlp.apache.org/
Penn Bio tokenizer	(Jin et al., 2006; R. McDonald & Pereira, 2005; R. T. McDonald et al., 2004)	http://www.seas.upenn.edu/~strctlrn/BioTagger/BioTagger.html
Stanford POS tagger	(Toutanova, Klein, Manning, & Singer, 2003)	http://nlp.stanford.edu/software/tagger.shtml
Specialist NLP	(Browne, Divita, Aronson, & McCray, 2003)	http://lexsrv3.nlm.nih.gov/LexSysGroup/Projects/textTools/current/Usages/Tokenizer.html
UIUC word splitter	-	http://cogcomp.cs.illinois.edu/page/tools_view/8
Xerox tokenizer	(Beesley & Karttunen, 2003)	http://open.xerox.com/Services/fst-nlp-tools/Consume/175

Table 3.3: Overview of the 21 tools reviewed in this chapter with their publications and website

Tool	Year	Version	Ref.	Install.	Src.	OS	Lang.	Depend.	License
Brill	1992	2004	1657	SA	✓	Any	Java	Java	-
DMT	1996	1996	-	SA	✓	Any	Perl	Perl	GNU GPL
ERG	2000	2011	273	SA	✓	Linux	Delph-in	Delph-in	GNU GPL
Freeling	2004	2012	200	SA	✓	Any	C++	C++/libboost, libicu libraries	GNU GPL
Genia	2004	2007	126	SA	✓	Unix	C++	C++	University of Tokyo
GUT	2002	2012	1480	SA	✓	Any	Java	Java	GNU LGPL/AGPL
Gump	-	-	-	SA	✓	Unix	Gump	Gump/OZ	-
JLT	2006	2007	23	SA	✓	Any	Java	Java	CPL
LingPipe	2011	2011	0	SA	✓	Any	Java	Java/Ant	It depends on the type of use
LT TTT	2000	2008	108	SA	✓	Linux/Mac OS X	LT-XML 2 tools	LT-XML 2 tools	University of Edinburgh GPL
Mallet	2002	2008	743	SA	✓	Any	Java	Java	CPL
MCP	2008	2013	69	SA	✓	Any	C++	C++	Apache license 2.0
MedPost	2004	2008	159	SA	✓	UNIX	C++/Perl	C++/Perl	United States Copyright Act
MXPOST	1996	1997	1445	SA	×	Any	Java	Java	Adwait Ratnaparkhi
NLTK	2009	2013	402	SA	✓	Any	Python	PyYAML/Pip/Python setuptools	Apache license 2.0
OpenNLP	-	2013	-	SA	✓	Any	Java	Java/Maven	Apache license 2.0
Penn Bio	2004	-	32	SA	✓	Any	Java	Java	-
Stanford	2003	2012	775	SA	✓	Any	Java	Java	GNU GPL
Specialist	2003	2006	47	SA	✓	Any	Java	Java	Open source resource
UIUC	-	2003	-	SA	✓	Any	Perl	Perl	University of Illinois
Xerox	2003	2013	586	API	×	Any	SOAP	-	Open Xerox

Abbreviations are as follows: '*Year*': year of the first publication of the tool. '*Version*': year when the tool was last updated. '*Ref.*': number of references found for the publication in Google Scholar (as of June 2013). '*Install.*': type of installation, i.e., stand-alone (SA) or application programming interface (API). '*Src.*': availability of source code. '*OS*': supported operating system. '*Lang.*': programming language in which the tool has been developed. '*Depend.*': dependencies on external software or libraries. '*License*': type of the license under which the tool is available.

Table 3.4: Comparison of all tools according to selected technical criteria

Tool	POS	Domain	Text reconstruction	Trainable	Integration
Brill	✓	General	✓	×	Java
DMT	×	General	✓	×	×
ERG	✓	General	✓	×	×
Freeling	✓	General	✓	×	C++
Genia	✓	Biomedical	✓	×	×
GUT	✓	General	✓	×	Java
Gump	×	General	✓	×	×
JLT	×	Biomedical	✓	✓	UIMA components
LingPipe	✓	General	✓	×	Java
LT TTT	✓	General	✓	×	×
Mallet	✓	General	✓	×	Java
MCP	✓	Biomedical	×	×	×
MedPost	✓	Biomedical	✓	×	×
MXPOST	✓	General	✓	✓	×
NLTK	✓	General	✓	×	Python
OpenNLP	✓	General	✓	✓	Java
Penn Bio	✓	Biomedical	✓	×	Java
Stanford	✓	General	×	×	Java
Specialist	✓	Biomedical	✓	×	Java
UIUC	✓	General	×	×	×
Xerox	✓	General	✓	×	SOAP

Abbreviations are as follows: '*POS*': part-of-speech tagger associated to the tool. '*Domain*': domain where the tokenizer was trained, i.e., general or biomedical. '*Text reconstruction*': Ability to convert the output of the tokenizer into the original text. '*Trainable*': Possibility to train the tool with other document collections. '*Integration.*': Possibility to integrate the tool in an application, i.e., if the tool has an application programming interface (API) and its programming language.

Table 3.5: Comparison of all tools according to selected functional criteria

Tool	Learning curve	Use	Install.	Doc.	Support	Start guide
Brill	◆	◆	◆	Enough	×	×
DMT	◆◆	◆◆	◆	×	×	×
ERG	◆◆◆	◆◆◆	◆◆	Poor	List/FAQ	✓
Freeling	◆	◆	◆	Very good	Forum/FAQ	✓
Genia	◆	◆	◆	Poor	×	✓
GUT	◆	◆	◆	Good	FAQ/List	✓
Gump	◆◆	◆	◆	Poor	×	✓
JLT	◆	◆	◆	Enough	Email	✓
LingPipe	◆	◆	◆	Fine	×	✓
LT TTT	◆◆	◆	◆	Poor	List	✓
Mallet	◆◆	◆◆	◆	Poor	List	✓
MCP	◆	◆	◆	Poor	×	×
MedPost	◆◆◆	◆◆	◆◆	Poor	×	✓
MXPOST	◆	◆	◆	×	FAQ	✓
NLTK	◆	◆	◆◆	Fine	List	✓
OpenNLP	◆	◆	◆	Good	List/Wiki	×
Penn Bio	◆	◆	◆	×	×	×
Stanford	◆◆	◆	◆	Enough	FAQ/List	✓
Specialist	◆◆	◆◆	◆◆	Poor	×	✓
UIUC	◆	◆	◆	×	×	×
Xerox	◆	◆	-	×	Forum	✓

Abbreviations are as follows: '*Learning curve*': worth looking at the tool (◆: easy, ◆◆: moderate, ◆◆◆: very difficult). '*Use*': easiness of use. '*Install.*': easiness of installation. '*Doc.*': Availability and quality of the documentation. '*Support*': Existence of support, i.e., mailing list, forum, FAQ or wiki. '*Start guide*': Availability of a start guide.

Table 3.6: Comparison of all tools according to selected usability criteria

3.3.4 Results of the selection phase

First, of the 21 tokenizers initially included and analysed based on the technical, usability and functionality predefined criteria detailed in Section 3.3.3, 8 of them are discarded from being tested on a subset of the BioScope corpus in a second phase, due to the fact that they show too many errors in the example tested by He and Kayaalp (2006). Other pitfalls according to the predefined criteria for each tokenizer are the following:

- Brill's POS tagger. It does not provide any support or start guide.
- Dan Melamed's tokenizer. It is difficult to use and it does not provide any support or documentation. In addition, it neither has an associated POS nor API.
- Gump tokenizer. It does not have any publication. It is written in Gump which could make its customisation difficult. It does not include an associated POS or API. The effort needed to learn the tool is moderate since it does not provide any support and the documentation is poor.
- LT TTT. Its documentation is reduced so the effort needed to learn the tool is not very easy. It does not include an API.
- Mallet tokenizer. Its ease of use is regular and its documentation is poor.
- MXPOST. Its binaries are not available.
- Specialist NLP. The easiness of installation and use is moderate and it does not provide any kind of support. Furthermore, the documentation is sparse.
- UIUC word splitter. It neither includes support nor documentation. It does not provide an API. It is not possible to reconstruct directly the input since it converts parentheses and squared brackets into their normalised tokens.

Secondly, the 13 remaining tokenizers (i.e., *English Resource Grammar*, *ERG*; *Freeling*, *Genia tagger*, *Gate Unicode tokenizer*, *JULIE LAB tokenizer*, *Lingpipe*, *McClosky-Charniak parser*, *MedPost*, *NLTK tokenizer*, *OpenNLP tokenizer*, *Penn Bio tokenizer*, *Stanford POS tagger* and *Xerox*) are tested on a set of 28 sentences extracted from the three sub-collections of the BioScope corpus. 50% of these sentences are papers, 28.5% are abstracts and 21.4% are clinical documents as shown Table 3.7. These sentences contain all the problematic cases described in Section 3.2 that may appear when a text is tokenized in general and in the biomedical domain in particular. This set of 28 sentences along with what should be the correct tokenization is described in the Appendix B.

Sub-collection	# Sentences	% Sentences	# Words	Length min. sentence (#words)	Length max. sentence (#words)	Av. Length sentences
Clinical	6	21.4	54	1	19	9
Papers	14	50	335	10	39	23.9
Abstracts	8	28.5	194	18	36	24.25
Total:	28		583			

Abbreviations are as follows: '*Min.*': minimum, '*Max*': maximum, *'Av.'*: average.

Table 3.7: Statistics about the 28 sentences from the BioScope corpus

An analysis of the errors introduced by each tool for the set of examples is described below. As errors have been counted the number of types of each mistake committed, independently of the number of times each of them occurred since it is not known how often each type of error happens in the biomedical literature. In any case, there is a strong positive correlation (Pearson's r value = 0.7131) between the total number of errors and the number of different types of error as can be found in Table 3.8.

It highlights that all tokenizers fail with chemical substance names, regardless of whether they are trained in the biomedical domain or not. This gives us an idea of the difficulty of tokenizing the biomedical terminology. Another pitfall is that they do not manage numeration correctly. In addition, some of them handle it inconsistently. Therefore, the efforts of developers should focus on improving these issues when building or customising new tokenization tools. Furthermore, more than 75% of the tokenizers fail with URLs, hypertext markup symbols and words with numbers and punctuation so these weaknesses should also be taken into account.

On the other hand, the strengths of the tokenization tools analysed include the correct treatment of the words with letters and brackets, words with letters and numbers, DNA sequences and temporal expressions. In fact, less than 38% of the tokenizers fail in one or more of these cases.

Error type	Tool and #ocurrences												
	ERG	Freeling	Genia	GUT	JLT	LingPipe	McClosky	MedPost	NLTK	OpenNLP	PennBio	Stanford	Xerox
Hyphenated compound words	3	1		5	8	7		8			7		
Letters and slashes		2	4				4		4	4		4	4
Letters and apostrophes			1	2		2	1	2	1	1			
Letters and brackets	3				1		1			1			
Letters and periods	1		1	1		1		1	1	1			
Letters and numbers				2							1		
Numbers and punctuation		2	1	9	6	5	6	9	2	2	4	2	
Enumeration	1	2	2	2	1	2	1	2	2	1	1	2	2
Hypertext markup symbol	1	2	2	2	2	2	2	2	2	2	2		2
URL		1	1	1	1	1	1	1	1	1	1		
Abbreviations and acronyms		9	2	13	8	10		10			9	8	
DNA sequence	1												
Temporal expressions				1	1	1		1			1		
Chemical substances	1	2	2	6	3	7	2	6	2	2	3	2	2
Others	15	4		2						1	7		3
Total number of errors	26	25	16	46	31	38	18	42	15	16	36	18	13
Total type of errors	8	9	9	12	9	10	8	10	8	10	10	5	5

Note that ERG tokenizer was tested on 15 of the 28 sentences.

Table 3.8: Number of errors per type and tool

Analysing the technical, functional and usability aspects of the 13 tokenization tools tested in this phase, it highlights the following:

- All the tools, except Xerox which is an API, have a stand-alone installation. In addition, most of them support any operating system.

- Only OpenNLP and JULIE LAB tokenizer give us the possibility of training the tool with another document collection. Most of the tokenizers are suitable to integrate in any real-world application and all of them provide associated POS tagging which is usually the following step in many NLP tasks.
- Among the tokenization tools with better documentation are Freeling, Gate Unicode Tokenizer and OpenNLP. They, together with JULIE LAB tokenizer, provide broader support.
- Most of them are easy to use. Only English Resource Grammar and MedPost show greater difficulty.

An in-depth analysis of the errors committed by each of the tools is shown below. In addition, the complete output of each tokenizer can be found in the Appendix C.

Despite being the tool with the largest number of references, **Gate Unicode tokenizer** is one of those which show more errors (to be exact 12 different errors). In fact, it fails in almost all the possible cases. It separates hyphenated compound words even when they indicate a substance name. It also fails with apostrophes except in words with single quotation. In addition, this tokenizer always separates the numbers or periods from the main word. It does not manage numbers in general, numerations, hypertext markup symbols, URLs, temporal expressions and abbreviations. It performs well with parentheses and brackets except when they are included in a substance name since it does not handle properly the complexities of the biomedical domain.

MedPost is one of the tools with a higher number of errors (10 in total). It fails with numbers, including ranges; percentages, fractions and so on. It also does not manage well hypertext markup symbols, numerations, temporal expressions, apostrophes, URLs and abbreviations. It is inconsistent with separated hyphenated compound words as can be seen in the example:

*Normal chest **x-ray** .*

*The patient had prior x - ray on **1 / 2** which demonstrated no pneumonia .*

This tokenizer fails with biomedical terminology despite being designed for this domain. An example could be the following:

*These results reveal a central role for CaMKIV/Gr as a **Ca(2+)-regulated** activator of gene transcription in T lymphocytes.*

Another aspect taken into account is that this tokenizer has not been updated since 2008 as well as it does not provide an API. In addition, this tool is really difficult to install and use and its documentation is poor.

The third and fourth tools that have been discarded are **Penn Bio tokenizer** and **LingPipe**. Both of them also show 10 errors. Penn Bio tokenizer is inconsistent in many cases. For instance, in words with letters and numbers as shown in the example below:

*Selenocysteine and pyrrolysine are the **21st** and **22nd** amino acids, which are genetically enc oded by stop codons.*

It is also inconsistent in hyphenated compound words as can be seen in the next example:

***2-year** 2-month* *old female with pneumonia.*

Other cases of inconsistency are ranges and numerations. In addition, this tokenizer fails with fractions, hypertext markup symbols, abbreviations, URLs, temporal expressions and percentages. It does not manage well biomedical terminology even when it is designed for this domain. It often splits some words into two such as in the following case:

*Of these, Diap1 has been most extensively **characteriz ed**; it can block cell death caused by the ectopic expression of reaper, hid, and grim (reviewe d in [26])*.

Furthermore, Penn Bio tokenizer could be difficult to use for people not familiarised with GATE especially when there is no documentation.

For its part, Lingpipe separates hyphenated compound words even in the case of biomedical terminology. It also fails with ranges, percentages, numerations, URLs, and temporal expressions, words with periods, hypertext markup symbols and abbreviations. It does not manage well biomedical terminology where it introduces several errors.

The last tool with 10 different types of errors is **OpenNLP tokenizer**. The main problem of this tool is its inconsistency with numerations. An example to illustrate this case would be the following:

2.

1 *Bioactivation of sulphamethoxazole (SMX) to chemically-reactive metabolites and subsequent protein conjugation is thought to be involved in SMX hypersensitivity.*

As the Genia tagger, OpenNLP tokenizer fails with hypertext markup symbols, URLs, percentages, words with periods and words with single quotation. It also never separates slashes from words and it fails with substance names when they include parentheses. Furthermore, it introduces some mistakes such as it does not separate square brackets from the word as well as no splitting words with a comma between them. For example:

Mutants in Toll cueing pathway were obtained from Dr. S. Govind: ***cactE8,cactIIIG*** *and cactD13 mutations in the cact gene on Chromosome II.*

Freeling, although it is easy to install and use and its documentation is really good, it shows too many errors compared with the best systems (to be exact 9 different mistakes). It fails in numerations and URLs. It removes hypertext markup symbols. In addition, it introduces the underscore symbol to separate certain words as the example below shows:

This was last documented on the Nuclearv_Cystogram dated 1/2/01.

This tokenizer also does not perform properly with abbreviations. A good illustration of this case could be the following:

The transcripts were detected in all the ***CD4- CD8-, CD4+ CD8+, CD4+ CD8-*** *and* ***CD4- CD8+*** *cell populations.*

Finally, it does not manage substance names well and words with slashes in some cases.

Genia tagger also shows 9 different errors despite being the most popular tool designed for the biomedical domain. It has not been updated since 2007. Among its failures it can be found that it never separates words with slashes even when they indicate alternatives or measurement units. It also fails in large numbers, percentages, numerations, URLs, hypertext

markup symbols, words with periods even when they denoted abbreviations and words with single quotation. In general, it manages biomedical terminology well. However, it fails with substance names which include parentheses as can be seen in the example:

> *These results reveal a central role for CaMKIV/Gr as a* ***Ca (2+)-regulated*** *activator of gene transcription in T lymphocytes.*

Another tool which has 9 different errors is **JULIE LAB tokenizer**. As many of the tokenizers described here, it handles numerations inconsistently. The same occurs with ranges. For instance, in the following sentence, the tokenizer separates the en dash symbol from the numbers:

> *E-selectin is induced within* ***1 - 2*** *h, peaks at* ***4 - 6*** *h, and gradually returns to basal level by 24 h.*

However, in the next one, it treats the range as a whole:

> *If both the best hits of the N - and C - terminal parts are statistically significant (E-value <* ***10-5*** *), and distance between them is less than 1 kbp, we call these hits ' syntenic hits '.*

Furthermore, it fails with numbers and punctuation except when they only include the period as punctuation mark.

Finally, it does not manage well URL because despite identifying correctly the boundaries of the URL, the tokenizer repeats the last letter. An example is the following:

> *Names of all available Trace Databases were taken from a list of databases at* ***http://www.ncbi.nlm.nih.gov/blast/mmtrace.shtml l***

McClosky-Charniak parser. Although it is constantly maintained and the number of errors is acceptable (8 different error types), this tool is inconsistent with numerations. In addition, it does not manage numbers such as *percentages* or *ranges* well and fails with hypertext markup symbols, URLs, brackets and names when it includes apostrophe. As Genia tagger and OpenNLP tokenizer, it never separates slashes from words as well as failing with substance names when they include parentheses.

NLTK tokenizer shows the same type of errors as the Genia tagger except that it does not fail with abbreviations.

The English Resource Grammar has been tested because there are no references about how it performs. All the sentences have not been tested on the online demo because it apparently fails due to the complexity of some of the examples. However, the amount of errors is significant in relation to the number of sentences processed (8 in total) and it does not perform well with biomedical terminology. In addition, it does not include an API and it is really difficult to learn how to use it. Therefore, this system is discarded.

The tools that show the least number of errors are the **Xerox tokenizer** and the **Stanford POS tagger** (both 5 mistakes). In addition, these tools are two of the most cited according to the number of references in Google Scholar. Xerox tokenizer and Stanford tokenizer never separate slashes from words so they fail with alternatives and measurement units. They also fail with numerations. They perform biomedical terminology well except when any parentheses are included. The Xerox tokenizer fails with hypertext markup symbols. The last type of error of this system is produced when it does not break the text into words that must be separated. An example could be the following:

*Mutants in Toll cueing pathway were obtained from Dr. S. Govind: **cactE8, cactIIIG**, and cactD13 mutations in the cact gene on Chromosome II.*

The Stanford tokenizer fails with percentages and abbreviations when they indicate substance names as can be seen in the example:

*The transcripts were detected in all the **CD4-CD8-, CD4+CD8+, CD4+CD8-** and **CD4-CD8+** cell populations.*

Stanford also has the particularity that it converts parentheses and squared brackets into the same normalised tokens (-LRB- or -RRB-).

The differences in number of errors between these tools are minimal. In addition, they are the tools which show better consistency and accuracy for the training set. Therefore, an evaluation has been carried out in order to determine which one performs best in the

tokenization of biomedical texts. A detailed description of the evaluation process can be found in the next section.

3.3.5 Results of the evaluation phase

Sentence splitting and tokenization performance is typically evaluated in terms of accuracy, i.e., the number of correct decisions divided by the total number of decisions being made (Tomanek et al., 2007b). Therefore, this measure is used in the evaluation. In the tokenization task, accuracy can be defined as the number of tokens correctly identified by the tokenizer divided by the total number of tokens in the sub-collection.

In this section, an in-depth analysis of the errors showed by each tokenizer is also conducted. In fact, common errors made by both systems are those listed below:

- They never separate words with slashes (even when they represent two options or measures).
- They fail with numerations.
- When a substance name includes parentheses, they fail by separating them.

For all the sub-collections, Xerox tokenizer fails with hypertext markup symbols (e.g., *<*). For its part, the Stanford POS tagger, fails separating the symbol *%* from the number as well as it splits some abbreviations such as *ER+*.

Based on the results obtained for each tool in the three different sub-collections tokenized, as it is described below, the results are competitive in all the cases.

In the clinical documents sub-collection, the results obtained by Stanford POS tagger (98.87%) are slightly higher than those obtained by Xerox tokenizer (98.54%). However, both of them achieve a great accuracy value. In fact, as can be found in Table 3.9, Stanford POS tagger only identifies correctly 16 tokens more than Xerox tokenizer of the total of 4,795 tokens or words in the sub-collection. By analysing the cases in which the Xerox tokenizer does not detect the tokens correctly, it can be found that the errors are due to the fact that the system separates many long words into two tokens (e.g., *lymphadenop athy*). Stanford

POS tagger fails to split some hyphenated compound words (e.g., *2-1/2 -␣year*) and it could be considered inconsistent in the treatment of this type of expression.

	Tokenizers					
	Xerox			Stanford		
	Sub-collection					
	Clinical	Papers	Abstracts	Clinical	Papers	Abstracts
Total of tokens	4,795	5,040	15,251	4,795	5,040	15,251
Correct tokens	4,725	4,959	15,013	4,741	5,027	14,994
Accuracy	98.54%	98.39%	**98.43%**	**98.87%**	**99.74%**	98.31%
Macro average		98.56%			**99.08%**	
Micro average		98.64%			**98.90%**	

Table 3.9: Accuracy of each tokenizer for the BioScope corpus

For the sub-collection of papers, as shown in the third and sixth columns of Table 3.9, the best accuracy is obtained by the Stanford POS tagger (99.74%). However, the results obtained by Xerox tokenizer (98.39%) are only slightly worse. These results are comparable to those obtained by a human performing the same task. In this sub-collection, both tokenizers fail when the last word of a sentence is capitalised because they do not separate the word and the endpoint. In addition, the Xerox tokenizer incorrectly joins some different words into one (e.g., *insteadof*) whereas the Stanford POS tagger separates some substance names (*thiol␣:␣protein*).

In the case of the abstract sub-collection, as can be found in the fourth and seventh columns of Table 3.9, both systems obtain a good accuracy value. As opposed to the previous cases, here, the Xerox tokenizer is the system which provides the highest performance. The accuracy value obtained by this tokenizer is 98.43% compared to 98.31% obtained by the Stanford POS tagger; consequently, the difference between these two systems should be considered non significant. Both tokenizers fail when the last word of a sentence is capitalised because they do not separate the word and full stop. Also as in the case of papers, the Xerox tokenizer joins some words into one (e.g., *invivo*). A new error introduced by this system is that it ignores the symbol ’. For its part, the Stanford POS tagger separates some words which included a hyphen (e.g., *-␣resistant*).

Macro and micro averages have been used to summarise the global results. Macroaveraging gives equal weight to each sub-collection, while microaveraging gives equal weight to each per-token identification. The values of these measures for both tokenizers are detailed in Table 3.9 as well. Attending to these results, which can be seen in the last two rows of the table and, as mentioned at the beginning of this section, accuracy obtained by the Xerox tokenizer and the Stanford POS tagger are really high (around 99%). There is no big difference between them which makes either of them suitable for the tokenization of biomedical texts. It is remarkable that neither these tokenizers which obtain the highest accuracy are trained in the biomedical domain. In addition, no differences are observed depending on the genre of the text, therefore making coherent the affirmations made by other authors regarding tokenization processes not explicitly related to the domain or the genre of the texts which are processed (Habert et al., 1998).

3.4 Conclusions and chapter summary

Tokenization is the segmentation of text into primary building blocks for subsequent analysis and it is considered the first step in NLP. Choosing the right tokenizer is a non-trivial task, especially in the biomedical domain, where it poses additional challenges, which if not resolved means the propagation of errors in successive NLP analysis pipeline. This chapter presents a comprehensive overview study of tokenization tools with the aim to provide a valuable guideline for NLP developers in the biomedical field to select the appropriate tokenizer as first phase of a text mining task. In addition, this contribution means, as far as we are aware, the first comparative evaluation carried out on tokenizers in the biomedical domain. The motivation of tackling this problem is detailed in Section 3.1.

All the biomedical domain difficulties, together with what is considered to be the correct tokenization in each of these difficult cases is described in Section 3.2. To choose the correct tokenization it has researched in depth into the literature and has followed the conventions, strategies and recommendations suggested by the authors.

Section 3.3 shows an overview of the tokenizers available in the literature and describes the selection and evaluation phases. In particular, Section 3.3.1 details the corpus used in the study, i.e., the BioScope corpus. The annotation process of a subset of sentences of this corpus

to carry out the evaluation of the two systems that show better features, accuracy and consistency in the selection phase, is also provided. Section 3.3.2 explains the process followed to create the list of tools for tokenizing texts to analyse. It also includes a description of the technical, functional and usability criteria employed to evaluate each of these tools. In Section 3.3.3, the 21 tools suitable for the tokenization of biomedical texts are surveyed based on technical, functional and usability criteria. In Section 3.3.4, after analyzing the 21 tools according to the criteria, 13 of them are tested on a set of 28 sentences from the BioScope corpus, which is a data set sufficiently representative of the tokenization problematic cases in the biomedical domain. Finally, Section 3.3.5 describes the evaluation on 1499 sentences from the three sub-collections of the BioScope corpus of the two tokenizers that show better features and more accuracy and consistency in the examples tested in the previous phase (Xerox tokenizer and Stanford POS tagger). The accuracy shown by these tokenizers is similar and very high in both cases, so they could be suitable for the tokenization of any biomedical text.

Chapter 4

Learning cues and their scope in the medical domain

4.1 BioScope corpus

The document collection used in this study is part of the BioScope corpus, which is described in Section 3.3.1 of this thesis. To be exact, the sub-collection used in the experiments consists of clinical documents since this contribution is part of the project described in de Buenaga et al. (2010). It contains 1,954 documents, each having a clinical history and an impression section in which the radiologist describes the conclusion or diagnosis obtained from the radiographies. Moreover, this sub-collection represents the major portion of the corpus and is the densest in negative and speculative cues. Specifically, 4.78% of the words in the sub-collection of clinical documents are negation or speculation keywords. In the sub-collection of papers, the percentage is 1.73%, whereas in the abstracts only 1.57% of the words are cues. As shown in Table 4.1, 6,383 sentences have been used, which contain 872 negation cues and 1,137 speculation keywords. The most frequent negative cue are *no* (77.0%), *without* (11.1%), and *not* (6.8%). In the case of speculative cues, the most common are *or* (22.5%), *may* (9.4%), and *evaluate for* (7.2%). Likewise, 6.15% of the words belong to the scope of any cue.

#Documents	1,954
#Sentences	6,383
#Words	42,495
#Negation cues	872
#Speculation cues	1,137
#Words in the scope of any negation cue	3,364
#Words in the scope of any speculation cue	5,336

Table 4.1: Statistics on the sub-collection of clinical documents in the BioScope corpus

4.2 Methodology

4.2.1 System architecture

To solve the problem of negation and speculation detection, the resulting system has been modelled as two consecutive classification tasks. They are implemented using supervised machine-learning methods trained on the annotated clinical documents from the BioScope corpus.

As shown in Figure 4.1, in the training phase, the data set is pre-processed to obtain a valid representation for the classification algorithm, both in the cue detection and the scope detection phases. In this representation and for the cue detection phase, each instance is a token of the clinical sub-collection which has a number of associated features. In the scope detection phase, an instance is a cue–token pair from the sentence.

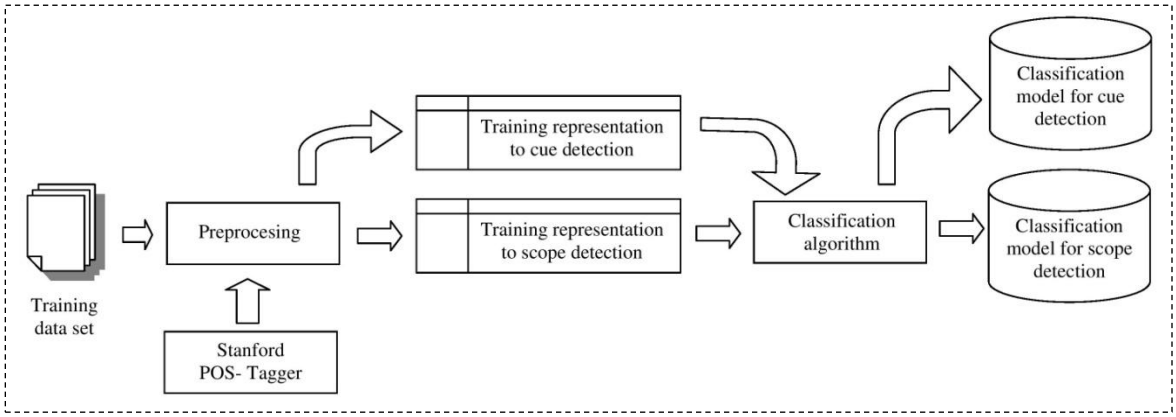

Figure 4.1: Training system architecture

Finally, the classification models for each of the two tasks are generated using different classification algorithms. In the test phase, the classification models obtained in the training phase are used to assess system performance. When the cues are detected, a classifier decides if the tokens in a sentence are at the beginning of a negation or speculation cue, inside or outside. This enables the system to find complex negation cues formed by more than one word. When the scope is detected, another classifier determines at sentence level the tokens affected by the cues previously identified. This means that, for every sentence that has negation or speculation cues, the classifier decides if the other words in the sentence are inside or outside the scope of the cue. The process is repeated as many times as cues appear in the sentence. Both phases are shown in Figure 4.2.

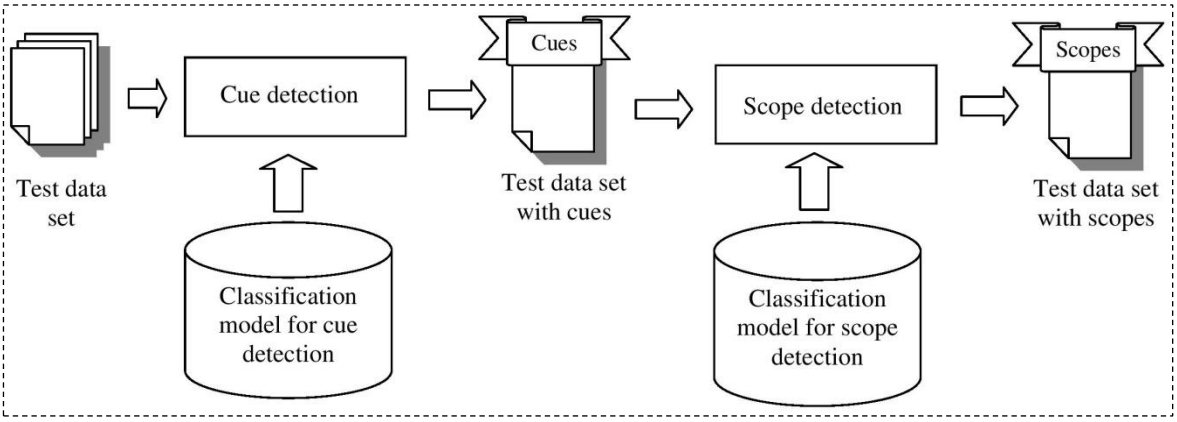

Figure 4.2: Whole system testing

The scope-finding system has also been tested using the gold-standard negation and speculation cues as shown in Figure 4.3.

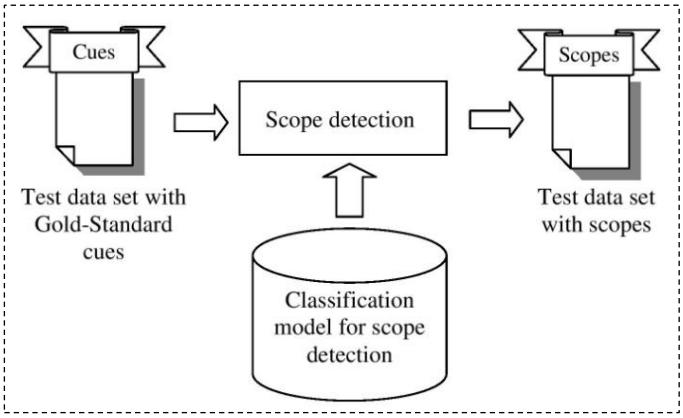

Figure 4.3: Testing the scope detection system

This is an imbalanced class problem, so it is considered that applying sampling techniques to the data could help solve the problem and improve the system performance. In this type of issue, the classification algorithms tend toward the majority class. Sampling techniques can solve this problem through an oversampling of the minority class and an undersampling of the majority class, using a random strategy. In this case, a supervised resample technique has been used in the scope detection phase. This technique produces a random sample data set using sampling with replacement. A 0 value in the class distribution parameter leaves the class distribution as it is, whereas a value of 1 ensures the class distribution is uniform in the output data. After experimenting with different class distribution parameter values, the value used has been 0.3 because it achieves the best performance. It has also been experimented with resample techniques in the cue detection phase, but they have not shown to be effective.

Naïve Bayes and C4.5 algorithms implemented in Weka (version 3.6) are used. Weka (Witten & Frank, 2005) is a popular machine-learning software suite that supports several standard data-mining algorithms. C4.5 is an algorithm for learning classification tasks that builds decision trees from a set of training data in the same way as ID3 (Quinlan, 1986), using the concept of information entropy (Quinlan, 1993). Decision trees are robust; they admit discrete and numerical values, and the splitting criterion (i.e., information gain) is fairly well established and accepted as good. Moreover, this method allows us to obtain the rules that explain the different ways of negation and speculation. In addition, García, Fernández and Herrera (2009) have shown how the approach of using sampling techniques with a C4.5 decision tree is highly competitive in terms of accuracy and is suitable for imbalanced problems. Another classifier with which it has been experimented is SVM as implemented in LIBSVM by Chang and Lin (2011). This classifier has been chosen over others because it has proven to be very powerful in text classification tasks where it often achieves the best performance, as described by Sebastiani (2002). Linear, Polynomial, Radial Basis Function (RBF), and Sigmoid kernels have been tested; LIBSVM has also been used to optimise the parameters g (gamma) and c (cost). As values to assess, those recommended by Hsu, Chang and Lin (2003) have been employed: $c = 2^{-5}; 2^{-3}; \ldots ; 2^{15}$ and $g = 2^{-15}; 2^{-13}; \ldots ; 2^{3}$.

The results obtained by each classifier are detailed in Section 4.3.

4.2.2 Features

The set of documents used in the experiments is organised into sentences. A sentence is a sequence of tokens each of which starts on a new line so it has been working at sentence level. To obtain the POS of tokens, the Stanford POS tagger has been used, which is a Java implementation of the tagger based on maximum entropy originally written by Toutanova and Manning (2000).

All tokens that appear in the sub-collection of clinical documents are represented by a set of features that are different in each of the two phases into which the task is divided. In both phases, it has been started with a large pool of selected features based on experience and previous works. These features encode information about the cue, the paired token, their contexts, and the tokens in between. The final feature set is obtained using the information gained and chi-squared feature selection techniques implemented in Weka on the initial set of attributes, starting with all the features and removing the least informative.

In the task of identifying negation and speculation cues, the instances are represented by the 13 features shown in Table 4.2. These features are divided into token level and token context level. All of the token context-level features are used in the case of the token to the left and the right of the token in focus. In that case, feature selection experiments show that the most informative feature is the lemma of the token, followed by the POS of the token and the lemmas and the POS of the tokens in context. Features such as a prefix that indicates if the token starts with *in* or *un* are irrelevant and therefore have been eliminated in the final set of attributes.

When the scope of the negative and speculative cues in the sub-collection is detected, it has been experimented with the following combination of features:

- Of the cue: Lemma, POS, and a tag that takes the value *NEG* if the cue is a negation cue or *ESP* if the cue is a speculation keyword.
- Of the paired token: Lemma, POS, and a tag that indicates if the paired token is inside or outside of the scope of the negation or speculation cue and which takes the value *ISN*, *ISE* or *OS*.
- Of the tokens between the cue and the token in focus: The distance in number of tokens and chain of POS types.

- Others: A tag that indicates the location of the token relative to the cue and that takes the value *PRE*, *INS* or *POST*. The lemma, POS, and type of the first token to the left and the first token to the right. The chain of types corresponding to the tokens between the cue and the token in focus. The cue number is divided by the total number of tokens in the sentence. The token number is divided by the total number of tokens in the sentence.

Token level	**Token context level**
Lemma	Lemma
POS	POS
Binary feature that indicates if a token is at the beginning of a sentence or not	Binary feature that indicates if a token is at the beginning of a sentence or not
Binary feature that indicates if a token is at the end of a sentence or not	Binary feature that indicates if a token is at the end of a sentence or not
Binary feature that indicates if a token is at the beginning of a document or not	Binary feature that indicates if a token is at the beginning of a document or not
Binary feature that indicates if a token is at the end of a document or not	Binary feature that indicates if a token is at the end of a document or not
Tag which takes the values *BN*, *IN*, *BE*, *IE*, *O* as if the token is at the beginning of a negation cue, inside a negation cue, at the beginning of a speculation cue, inside a hedge cue or outside	

Table 4.2: Features in the task of identifying negation and speculation cues

The most informative features in scope detection, according to the feature selection experiments, are the information about the cue (i.e., the lemma of the cue, followed by the tag that indicated if the cue is a negation or speculation and the POS of the cue). In that case, information about whether the cue or the paired token is at the beginning or at the end of a document or sentence is the least informative. Besides removing less significant features from the final set of features, different combinations of attributes have been tested. They show worse performance than that of the combination explained above.

4.2.3 Post-processing rules

In the cue detection phase, the F_1 shown by the system for some speculation cases has been obtained applying a post-processing. Its pseudo code is shown in Figure 4.4. When the cue is formed by a single token, the algorithm changes the class of tokens classified as inside a cue for the start of a keyword. When the cue is formed by more than one token and the different types tokens in the cue does not match, i.e., some have been classified as speculation and others as negation, the ratings given to that expression so far is consulted and the class is replaced by that with the highest appearance frequency. If it has not been classified yet, the class is replaced as speculation since the number of hedge cues is greater than the negation keywords in both data sets.

```
IF length(signal)=1
       IF signal.type=”in” THEN
              signal.type=”bn”
       ENDIF
       IF signal.type=”ie” THEN
              signal.type=”be”
       ENDIF
       IF signal doesn’t appears in test dictionary THEN
              dictionaryTest.add(signal)
       ENDIF
       IF signal appears in test dictionary THEN
              increase frequency
       ENDIF
ENDIF
IF NOT
       IF all words inside the signal don’t have the same type THEN
              Find signal in train dictionary classified as negation
              Find signal in train dictionary classified as speculation
              IF NOT appears THEN
                     Find signal in test dictionary classified as negation
                     Find signal in test dictionary classified as speculation
                     IF appears THEN
                            Compare frequencies and change the type of the signal for
                            those with higher frequency
                            Increase frequency of those with higher frequency in test
                            dictionary
                     ENDIF
                     IF NOT appears THEN
                            Change the type of the signal for speculation
                            dictionaryTest.add(signal)
                     ENDIFNOT
              ENDIFNOT
              IF appears THEN
                            Compare frequencies and change the type of the signal for
                            those with higher frequency
                            Increase frequency of those with higher frequency in
                            train dictionary
              ENDIF
       ENDIF
ENDIFNOT
```

Figure 4.4: Cue detection post-processing algorithm pseudo code

As in the case of cue detection, some of the results obtained in the scope recognition phase, have been obtained by applying a simple post-processing algorithm on the output of the classifier. This algorithm removes the scope consisting of a single token. If a token is classified as belonging to the scope of a cue but the word on the left and right is outside the scope, the algorithm changes the type of the token as not belonging to the cue. Figure 4.5 shows the pseudo code of this post-processing algorithm.

```
i<-2

WHILE i<length(words) DO

      IF words(i-1).type=”os” AND words(i+1).type=”os” AND
      words(i).type<>”os” THEN

words(i).type<-“os”

      ENDIF

      i<-i+1

ENDWHILE
```

Figure 4.5: Scope detection post-processing algorithm pseudo code

4.3 Results

The aim of the system previously described is to identify negation and speculation cues and their scope in clinical documents. The results are obtained by training and evaluating the system with the sub-collection of clinical reports of the BioScope corpus. Specifically, the sub-collection has been randomly divided into three parts and 2/3 are used to train and 1/3 to evaluate.

The results are compared in different ways.

4.3.1 Evaluation and measures

To obtain the system performance, two different tests have been carried out: token level evaluation and cue level evaluation. In both cases, in the negation and speculation cue detection task, a token is correctly identified if it has been classified as being at the beginning, inside, or outside the negation or speculation cue. Precision, recall and their harmonic mean F_1-score (Rijsbergen, 1979) are used as measures.

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

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

$$F_1 = \frac{2PR}{P+R}$$

In the token level evaluation, within the task of identifying the scope, a token is correctly classified if it has been properly classified as being inside or outside the scope of all negation or speculation cues that appear in the sentence. This means that if there is more than one negation or speculation cue in the sentence, the token is correctly assigned a class for each of these cues. The evaluation takes the token as a unit. The same measures as in the cue detection task have been employed. In this case:

$$\text{Precision (P)} = \frac{\#\text{tokens belonging to some scope correctly identified by the system}}{\#\text{tokens belonging to some scope identified by the system}}$$

$$\text{Recall (R)} = \frac{\#\text{tokens belonging to some scope correctly identified by the system}}{\#\text{tokens belonging to some scope in the test collection}}$$

F_1 is calculated using the same expression as in the cue detection task.

On the other hand, also in the scope recognition task, the percentage of scopes correctly classified (PCS) is evaluated. This is a cue level evaluation and therefore takes the cue as a unit. In this case, the scope associated with a cue is correct when all the tokens of a sentence have been correctly classified as inside or outside the scope of the cue.

Finally, note that in both evaluations, negation and speculation have been assessed separately.

4.3.2 Cue detection results

As detailed in Section 4.2.1, it has been experimented with C4.5 classifier and SVM classifier. With the latter, experiments with the main types of kernels have been carried out, optimising in each case the parameters *c* and *g*. Table 4.3 shows the results obtained by these classifiers in the cue detection phase.

For reasons of space and clarity in the tables, the results obtained after applying the post-processing algorithm are shown only in cases where this process improves the initial results.

	Negation			**Speculation**		
	Precision	Recall	**F_1**	Precision	Recall	**F_1**
C4.5 classifier	96.5	98.0	**97.3**	92.1 (92.8*)	92.1(93.4*)	92.4(93.1*)
SVM classifier Linear $c=2, g=2^{-11}$	96.5	97.4	96.9	94.3(94.5*)	93.2(93.6*)	93.7(94.1*)
SVM classifier Polynomial $c=2^{-3}, g=2^{-1}$	97.3	93.9	95.6	95.5	80.6	87.45
SVM classifier RBF $C=2^{5}, g=2^{-5}$	96.8	97.1	96.9	95.9	93.2	**94.9**
SVM classifier Sigmoid $c=2^{7}, g=2^{-7}$	96.8	97.1	96.9	95.4(95.4*)	93.2(93.4*)	94.3(94.4*)

* Result obtained after applying the post-processing algorithm

Table 4.3: Performance of negation and speculation cue detection of C4.5 classifier and SVM classifier, in terms of Precision, Recall and F_1 (%)

Although the results obtained in the negation cue detection task are slightly higher than those obtained in speculation, all the algorithms in general obtain a great performance value. The best F_1 in negation, as shown in the third column, is obtained by C4.5 classifier (97.3%). However, the difference with the results achieved by the best model of SVM is not significant (97.3% versus 96.9%). In the speculation detection task, SVM classifier with RBF kernel yields a 94.9% of F_1. That is the best result although, as mentioned above, in general all the

classifiers presented a good performance value. Only SVM classifier when Polynomial kernel is used obtains a lower F_1 value than the best.

Six different systems are employed to compare the performance of the proposed system. Two baseline algorithms are used. The first baseline is created by tagging the 2 most frequent expressions of negation and speculation in the training data set as cues. In the second baseline, the 8 most frequent expressions are used. Likewise, in the case of negation, the results are compared with those obtained by NegEx for the same test data set. The comparison for speculation detection with NegEx cannot be performed because this system has not been designed to detect these types of cues. Another system with which the results yielded by the proposed system are compared is that developed by Morante and Daelemans (2009a; 2009b). This system is very efficient both in negation and speculation detection and the results have been obtained by the authors by training on the full abstract sub-collection and testing on clinical sub-collection, both from the BioScope corpus. The works developed by Zhu et al. (2010) and Velldal et al. (2012) follow the same experiments and evaluation scheme. Although the results obtained by this latter is considered as the state-of-the-art, comparison in speculation is not possible because they do not test their system on the clinical subcollection.

Table 4.4 shows the results for these four classifiers and the baseline systems.

	Negation			**Speculation**		
	Precision	Recall	**F_1**	Precision	Recall	**F_1**
Baseline 1	98.1	85.9	91.6	97.2	33.6	50.0
Baseline 2	96.5	98.4	97.4	94.9	70.5	80.9
NegEx	63.9	67.4	65.6	-	-	-
Morante	100	98.0	**99.0**	71.2	52.3	60.3
Zhu '10	88.5	86.8	**87.6**	91.7	33.3	48.9
Velldal	96.4	95.9	**96.1**	-	-	-
Our system	96.5	98.0	97.3	95.9	93.2	**94.9**

Table 4.4: Performance of negation and speculation cue detection of the proposed classifier, baseline algorithms, NegEx and the systems developed by Morante, Zhu and Velldal in terms of Precision, Recall and F_1 (%). The proposed system uses a C4.5 classifier for the negation cue detection and a SVM RBF classifier for the speculation cue detection

In the case of negation detection, as shown in the fourth column, the first baseline already obtaines a reasonably good performance value. This is because the two most frequent expressions of negation represent 88.1% of all expressions of negation present in the training data set. This does not happen in the case of speculation, where, as shown in the last column, the performance is lower. In this case, the two most frequent expressions of speculation represent only 31.9% of the total. The second baseline system shows how with a more comprehensive list of cues it is possible to improve the performance values obtained by the first baseline. NegEx, for its part, returns the worst results in negation detection. This may be because the system is not specially designed to work with documents from the radiology domain. As shown in the fourth column, the system developed by Morante and Daelemans (2009b) achieves the best F_1 value (99.0%) in negation detection. However, the difference with the F_1 value obtained by the second baseline (97.4%), the C4.5 (97.3%) or SVM (96.9%) classifiers and by the system developed by Velldal et al. (96.1%) is minor. In all cases, the result would be comparable to those obtained by a human rater performing the same task.

Speculation detection, as shown the F_1 values obtained by the baseline algorithms, is more complicated since the most frequent cues are not concentrated in a small number of expressions, as in the negation recognition. In this case, as shown in the last column of the table, the difference between all systems in terms of precision, recall and F_1 is relevant. The SVM RBF classifier provides the highest performance. The F_1 value obtained by this algorithm is 94.9% compared to 60.3% obtained by the system developed by Morante and Daelemans (2009a) and to 48.9% achieved by the method proposed by Zhu et al. (2010). These systems presented a low value of F_1 which is even lower than that reached by the second baseline (80.9%). The F_1 yielded by Zhu et al. (2010) is also lower than the F_1 obtained by the first baseline. The difference between C4.5 and SVM RBF classifier is not relevant (94.9% versus 93.1%) as shown in Table 4.4.

Therefore, in the case of negation detection, all systems except NegEx achieve high performance values. In terms of speculation detection, the SVM RBF classifier obtains the best results and there is a significant difference compared with the system developed by Morante and Daelemans (2009a) and Zhu et al. (2010).

4.3.3 Scope detection results

In the scope detection phase, the results obtained by C4.5 and SVM classifiers are reported in Table 4.5 and Table 4.6. As in the cue recognition phase, some of the results are achieved after applying a postprocessing algoritmh on the output of the classifier. Only results improved by this algoritm are shown.

Gold-standard cues

	Negation				**Speculation**			
	Precision	Recall	**F_1**	PCS	Precision	Recall	**F_1**	PCS
C4.5 classifier	91.7	88.7	90.2	**91.3**	80.1(81.8*)	70.9(70.0*)	75.3(75.5*)	55.7(58.6*)
SVM classifier Linear $c=2^{-1}, g=2^{-1}$	93.3	90.7	92.0	89.4	89.6	68.1	77.4	56.4
SVM classifier Polynomial $c=2^{3}, g=2^{-3}$	94.1	91.8	92.7	87.8	87.8	75.0	**80.9**	**68.4**
SVM classifier RBF $c=2^{15},g=2^{-5}$	93.8	92.7	**93.2**	89.4	89.9	72.3	80.1	67.4
SVM classifier Sigmoid $c=2^{11}, g=2^{-13}$	93.0	91.5	92.2	90.3	89.9	67.8	77.3	58.6

* Result obtained after applying the post-processing algorithm

Table 4.5: Performance of scope detection of C4.5 classifier and SVM classifier in terms of Precision, Recall, F_1 and PCS with gold standard cues (%)

As it occurrs in the cue detection experiments and due to the complexity of the speculation detection task, the results for speculation are worse than those obtained in negation. In that case, the SVM classifier outperforms the results obtained by the C4.5 classifier. The F_1 achieved by the SVM Polynomial classifier is 80.9% versus the 75.5% achieved by the C4.5

classifier. In terms of PCS, the difference between both classifiers is important because the result obtained by theC4.5 classifier is 58.6% and the result yielded by the SVM Polynomial classifier is 68.4%. These values are shown in the fourth and fifth columns of Table 4.5. The results are competitive in negation but improvable in speculation, especially in terms of PCS.

With predicted cues, as shown in Table 4.6, in the negation detection, the differences between the classifiers are not significant both in terms of F_1 and in terms of PCS measure. The C4.5 classifier identifies 89.2% of the full scopes correctly, whilst the SVM RBF classifier correctly recognises 87.8%.

Predicted cues

	Negation				**Speculation**			
	Precision	Recall	**F_1**	PCS	Precision	Recall	**F_1**	PCS
C4.5 classifier	89.6	85.8	87.7	**89.2**	73.2	58.9	65.3(65.4*)	49.5(51.9*)
SVM classifier Linear $c=2^{-1}, g=2^{-1}$	91.8	88.3	90.0	87.2	83.2	57.7	68.1	50.9
SVM classifier Polynomial $c=2^{3}, g=2^{-3}$	92.8	88.0	90.3	86.9	84.5	54.0	65.9	62.1
SVM classifier RBF $c=2^{15},g=2^{-5}$	92.1	89.7	**90.9**	87.8	84.8	62.5	**71.9**	**62.9**
SVM classifier Sigmoid $c=2^{11}, g=2^{-13}$	91.6	86.6	89.1	87.6	83.9	49.6	62.3	52.8

* Result obtained after applying the post-processing algorithm

Table 4.6: Performance of scope detection of C4.5 classifier and and SVM classifier in terms of Precision, Recall, F_1 and PCS with predicted cues (%)

In speculation detection using predicted cues, the results obtained by the SVM RBF classifier are higher than those of the C4.5 classifier. In terms of F_1 the results yielded by the C4.5

classifier are (65.4%) whilst those obtained by the SVM RBF classifier are (71.9%). The difference in PCS measure is of 11%; SVM RBF classifier obtains 62.9% against 51.9%. This difference is due to errors that the C4.5 classifier accumulates in the scope detection where its F_1 is significantly lower than the F_1 reported by the SVM classifier.

For this task, the performance of the proposed system are compared with the results obtained by Morante and Daelemans (2009a; 2009b), Zhu et al. (2010; 2013) and Velldal et al. (2012) as shown in Table 4.7. The evaluation of the system carried out by these authors is the same as in the cue detection task, i.e., training the system on the whole abstract sub-collection and testing it on the clinical sub-collection. The comparison has been done in two ways: using as cues those which appear directly in the documents (i.e., gold-standard cues) and using the cues that the system has identified in the previous phase (i.e., predicted cues). In addition, is important to notice the following:

- As in the cue detection phase, the results obtained by Velldal et al. (2012) can only be compared in negation. They show the results in terms of F_1 with gold-standard cues and in terms of Precision, Recall and F_1 with the predicted ones.
- Zhu at al. (2013) test their system with the gold-standard cues so comparison for the whole system is not possible. The results provided are in terms of PCS.
- Zhu et al. (2010) show the results in terms of PCS in the case of the gold-standard cues and according to Precision, Recall and F_1 for the predicted cues.

With gold-standard cues, all the systems have similar performance measures. In the case of negation detection, the systems are efficient. These results are shown in Table 4.7. The proposed system obtains a higher value of F_1 (93.2%) than the systems developed by Morante and Daelemans (2009b) and Velldal et al. (2012) which yields 92.0% and 91.4%, respectively. In terms of PCS, the developed approach correctly identifies more full scopes (89.4%) than those recognised by Morante and Daelemans (87.2%) and Zhu et al. (2013) who identify 85.3%. However, it does not happen the same compared with the results obtained by Zhu et al. (2010) whose system correctly determines 89.7% full scopes.

As in the cue detection, the results for speculation are worse. In this case, the performance can be improved, especially in the PCS measure where the proposed method achieves a value

of 68.4%, Morante and Daelemans (2009a) obtain a value of 60.5% and Zhu et al. (2010) yield 68.7%. Only Zhu et al. (2013) achieve a slightly higher value (72.9%).

Gold-standard cues

	Negation				**Speculation**			
	Precision	Recall	F_1	PCS	Precision	Recall	F_1	PCS
Morante	91.6	92.5	92.0	87.2	79.1	78.1	78.6	60.5
Zhu '10	-	-	-	89.7	-	-	-	68.7
Zhu '13	-	-	-	**85.3**	-	-	-	**72.9**
Velldal	-	-	91.4	-	-	-	-	-
Our system	93.8	**92.7**	93.2	**89.4**	**87.8**	**75.0**	**80.9**	**68.4**

Table 4.7: Performance of scope detection of the proposed classifier and the systems developed by Morante, Zhu and Velldal in terms of Precision, Recall, F_1 and PCS with gold standard cues (%). The developed system consists of SVM RBF classifier in negation detection and SVM Polynomial classifier in speculation detection.

With predicted cues, in the negation detection, the differences among the systems are not significant, as Table 4.8 shows. However, in terms of PCS measure, the developed system identifies 84.2% of the full scopes correctly, whilst the system developed by Morante and Daelemans (2009b) correctly determines 70.7%.

In speculation detection, the results obtained by the developed system are considerably higher than all other systems. In terms of F_1, the results yield by the proposed method double the values of the others. It achieves a value of 71.9% in F_1 while Morante and Daelemans and Zhu et al. (2010) obtain values of 38.1% and 35.7%, respectively. The difference in PCS measure is greater, specifically 62.9% against the 26.2% achieved by Morante and Daelemans.

These results show that the developed system is comparable with competitive systems and in some cases may even surpass the results obtained by these authors, improving the state-of-the-art results. This is especially important when evaluating the whole system, where the proposed method correctly identifies around twice as many scopes associated with speculation cues as that of the Morante and Daelemans' system.

	Predicted cues							
	Negation				Speculation			
	Precision	Recall	F_1	PCS	Precision	Recall	F_1	PCS
Morante	86.3	82.1	84.2	70.7	68.2	26.4	38.1	26.2
Zhu '10	82.2	80.6	81.4	-	70.4	25.5	37.5	-
Velldal	89.6	89.4	89.5	-	-	-	-	-
Our system	92.1	**89.7**	90.9	**87.8**	**84.8**	**62.5**	**71.1**	**62.9**

Table 4.8: Performance of scope detection of the proposed classifier and the systems developed by Morante, Zhu and Velldal in terms of Precision, Recall, F_1 and PCS with predicted cues (%).The proposed system consists of SVM RBF classifier both in negation and in speculation

4.3.4 Error analysis

4.3.4.1 Cue detection

The cases in which the system does not detect the cues correctly, could be classified into two types: (a) false-positive errors in which the system identifies as cues words that are not marked as keywords in the sub-collection and (b) false-negative errors in which the system does not recognise as cues words that are marked as such in the sub-collection.

The first category of errors, in negation cue detection, is observed in 11 cases out of the 313 negation cues presented in the test sub-collection. This represents an error rate of 3.51%. Most of these errors occur because speculation cues that include the words *not* or *no* appear in the sub-collection. For example, the keyword *no evidence* is always marked as a speculation cue in the sub-collection. Each time this cue appears, the proposed system identifies the word *no* as a negation cue. This is because this word appears 433 times in the training sub-collection identified as a keyword compared to five occurrences of *no evidence*.

In speculation cue detection, there are 17 cases of error. The total number of speculation cues in the test sub-collection is 428; therefore, the error rate is 3.97%. Errors in this case arise mainly because the system recognises as cues some words that appear in the training collection mostly classified as such. However, in the test collection, in some cases, these words are not keywords. For example, the cue *could* is marked in the sub-collection 38 times

as a speculation cue, but twice it is not marked. In these two cases, the developed system classifies the words as a speculation keyword.

The second type of error, in negation cue detection, occurs in 6 cases out of the 313 negation cues presented in the test sub-collection. The error rate is 1.91%, slightly lower than in the case of false-positive errors. These errors occur because some keywords are always marked as speculation cues except in one case, in which it is marked as a negation cue. For example, this occurs with the keywords *may* or *rule out*. The first is marked as a speculation cue 66 times and the second one 37 times in the train sub-collection. Obviously, in the case where these keywords are marked as negation cues, the proposed system fails and classifies them as speculation cues. In speculation, the errors occur 26 times. The error rate is 6.07%, higher than in the other type of errors. In this case, errors are mostly of two types. One consists of cues that include the words *no*/*not*/*cannot* and the proposed system classified only the words *no*/*not*/*cannot* as a negation cue (this type of error is the same as the false-positive errors in the negation detection described above). The other type of error occurs because the system does not identify as speculation cues those expressions that have infrequent occurrences. An example is *maybe*, which only appears twice in the train sub-collection and each time is marked as a speculation keyword. Here, the developed system does not detect these cues as such.

4.3.4.2 Scope detection

The most frequent errors occurred when the system identifies the scope of the cues can be divided into a wide range of categories, as shown in Figure 4.6.

1) The beginning of the scope is correct, but the system incorrectly extends the scope beyond the end of the sentence. For example, in the phrase *viral or reactive airways disease*, the scope of the cue *or* is the words *viral* and *reactive*, whereas the system recognises as scope these words but also *airways* and *disease*. For negation, this type of error represents 42.4% of the total; for speculation, it is 45.4%.

2) The scope identified by the developed system begins after the correct scope and is extended beyond the end of the sentence. For example, the scope of the cue *or* in the phrase *considerations include community acquired or atypical pneumonia such as mycoplasma* is formed by the words *community*, *acquired*, *atypical*. However, the

system recognises the words *acquire*, *atypical, pneumonia, such, as*, *mycoplasma*. This type of error comprises 3.8% of the total in negation and 7.5% in speculation.

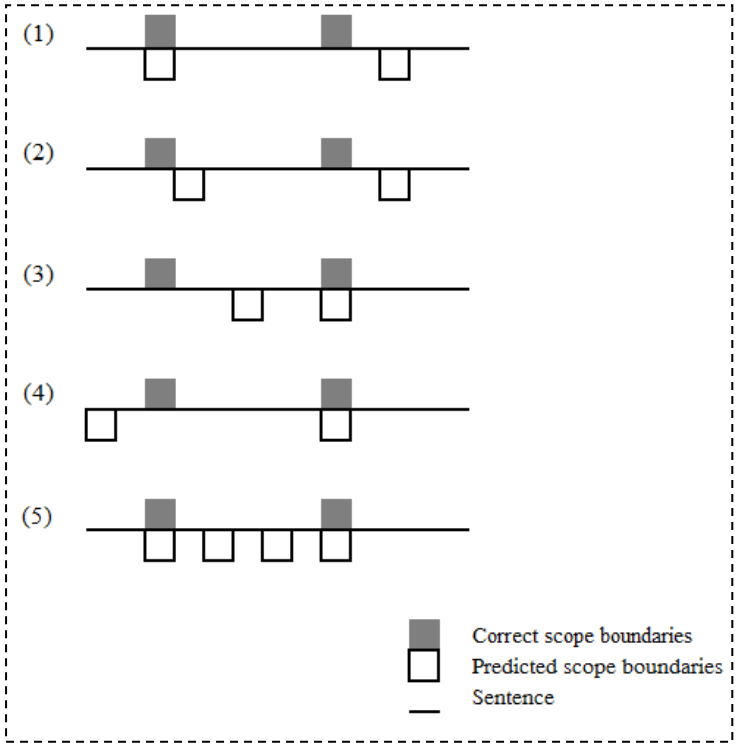

Figure 4.6: Errors in scope detection task

3) The end of the scope is correct but it begins after the correct position. For example, in the sentence *This may represent areas of atelectasis and/or pneumonia*, the system identifies the word *pneumonia* as scope while the correct scope is formed by the words *areas*, *of*, *atelectasis*, *pneumonia*. This error represents 21.5% of the total in negation and 13.6% in speculation.

4) The scope determined by the system is correct, except that it includes words that appear before the cue. This type of error occurs in 8.3% of negation cues and in 10.6% of speculation keywords. An example would be the sentence *There is no focal, lobar consolidation to suggest bacterial pneumonia*. The correct scope is *bacterial pneumonia*, but the proposed system includes also the word *focal* in the scope.

5) The scope identified by the system begins and ends correctly, but the system does not recognise as belonging to scope all the words that compose it (i.e., it incorrectly omits some words). This error represents 8.3% of the total for negation and 6.0% for speculation. For example, the system determines as scope the words *viral reactive airways disease* instead of *viral small airways reactive airways disease* (in this case, it omits exactly two words, *small* and *airways*).

4.4 Conclusions and chapter summary

This chapter describes a machine-learning system that identifies the negation cues and their scope in clinical texts. This contribution highlights by the fact that the proposed method improves the results to date for the sub-collection of clinical documents of the BioScope corpus. This sub-collection is detailed in Section 4.1.

The Section 4.2 shows the methodology applied to solve the problem. The system architecture is presented in Section 4.2.1. Basically, it consists of two consecutive classification tasks implemented using supervised machine-learning methods trained on the annotated documents previously mentioned. All tokens that appear in the collection of documents used for the experimentation are represented by a set of features which are different in each of the two phases into which the task is divided. They are explained in Section 4.2.2. With the aim of improving the results, the output of the classifier for both the cue and scope detection tasks, have been modified according to post-processing algorithms. The pseudo-code of these algorithms is detailed in Section 4.2.3.

The proposed system presents original aspects compared to previous research. In a simple architecture, SVM has been used as classifier algorithm due to the fact that it has proven to be very powerful in text classification task as well as it has hardly been employed to solve this task. Different kernels have been tested and its parameters have been optimised.

This is an imbalanced class problem. Supervised resample techniques have been used showing that applying sampling techniques to the data help solve the problem and improve the system performance.

New features, such as the place of the cue in the sentence, the distance between the cue and the token in focus, etc., have been explored.

The Section 4.3 describes and discusses the results. First, the measures used to evaluate the performance of the system are explained in Section 4.3.1. Next, Sections 4.3.2 and 4.3.3 show the results for the cue detection and scope recognition, respectively. These results show the superiority of the machine-learning-based approach regarding the use of regular expressions. In the detection of negation expressions, the developed system improves the F_1 of NegEx by 30%. In speculation, the proposed method beats the F_1 of the best system by almost 10%. Moreover, the results are compared with those obtained by the machine-learning system developed by Morante and Daelemans (2009a; 2009b). In the case of negation scope detection, the developed global system correctly determines approximately 20% more than the scopes identified by the other system. In speculation, this difference is greater and the proposed method correctly recognises nearly twice the number of scopes. Finally, an error analysis is provided in Section 4.3.4. Exactly, the errors introduced in the cue detection phase are described in Section 4.3.4.1 while the most common mistakes encountered in the scope recognition phase are detailed in Section 4.3.4.2.

Chapter 5

Learning cues and their scope in review texts

5.1 SFU Review corpus

5.1.1 Annotation process

The Simon Fraser University (henceforth, SFU) Review corpus (Taboada, 2008) has been chosen for the annotation of negation and speculation. This corpus is extensively used in opinion mining (Rushdi Saleh, Martín-Valdivia, Montejo-Ráez, & Ureña-López, 2011; Taboada et al., 2011; Martınez-Cámara, Martın-Valdivia, Molina-González, & Urena-López, 2013) 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 this one where all 400 documents are annotated at the token level with negative and speculative cues, and at sentence level with their linguistic scope (Konstantinova et al., 2012). The entire corpus has been annotated by one linguist adapting the existing Bioscope corpus guidelines (Szarvas et al., 2008) in order to fit the needs of the review domain. A second linguist has been annotated 10% of the documents, randomly selected and in a stratified way, with the aim of measuring inter-annotator agreement.

The annotation indicates the boundaries of the scope and the cues, as shown in (3) below. In the annotation, scopes are extended to the largest syntactic unit possible and the cues are never included in the scope.

(3) 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 keywords whose scope is not indicated (4.62% of the total of cues).

The exhaustive annotation guidelines followed in the annotation process together with the inter-annotator agreement analysis are described in Konstantinova and de Sousa (2011); Konstantinova et al. (2012).

5.1.2 Corpus characteristics

Table 5.1 summarises the main characteristics of the SFU Review corpus which is used by the system presented in this chapter as a learning source and for evaluation purposes.

	#Documents	#Sentences	#Words	Av. length documents (in sentences)	Av. length documents (in words)	Av. 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**

'Av.' stands for average.

Table 5.1: Statistics about 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 just over 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 event level so it cannot be compared to the SFU Review corpus. The last columns in the table show that there are important differences in the length of the documents depending on the domain but not in the length of sentences, which suggests that sentence complexity in the entire corpus is comparable.

In the case of negation, out of the total number of 17,263 sentences, 18% contain negation cues as shown in Table 5.2. 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. The most frequent negation cues are *not* (40.23%) and *no* (14.85%) which constitute more than 55% of the total frequency of all the negation cues found in the corpus. In addition, 5.85% of the words belong to the scope of any of these cues, most of which are extended to the right (99.40%) as the last row of the table shows. Only 0.93% of the scopes are extended to the left of the negation word.

In the case of speculation, as Table 5.3 shows, 22.7% of the total of sentences is speculative. This proportion is higher than the negative sentences because of the nature of the corpus, where speculation is widely used to express opinions. By comparison, the BioScope corpus has fewer than 20% of the sentences as speculative. *If* (16.34%), *or* (15.30%) and *can* (14.27%) are some of the most frequent speculation cues. Nevertheless, they do not represent the majority of the speculation cases, as reported for negation. The number of occurrences of each cue however, was equally distributed across all the documents. Likewise,

12.05% of the words belong to the scope of some cue. In this case, as the two last rows of the table show, 99.79% of the scopes are extended to the right and 17.54% to the left.

	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
Av. 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
Av. 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
Av. 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

'*Av.*' stands for average.
Av. 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 5.2: Negation 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
Av. 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
Av. 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
Av. 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

Same notes as in Table 5.2 apply

Table 5.3: Speculation statistics in the SFU Review corpus

5.2 Methodology

5.2.1 System architecture

The *identification of negation and speculation cues and the determination of their scope* are modeled as two consecutive classification tasks (see Figure 5.1). They are implemented using supervised machine learning methods trained on the SFU Review corpus (Konstantinova et al., 2012)[6].

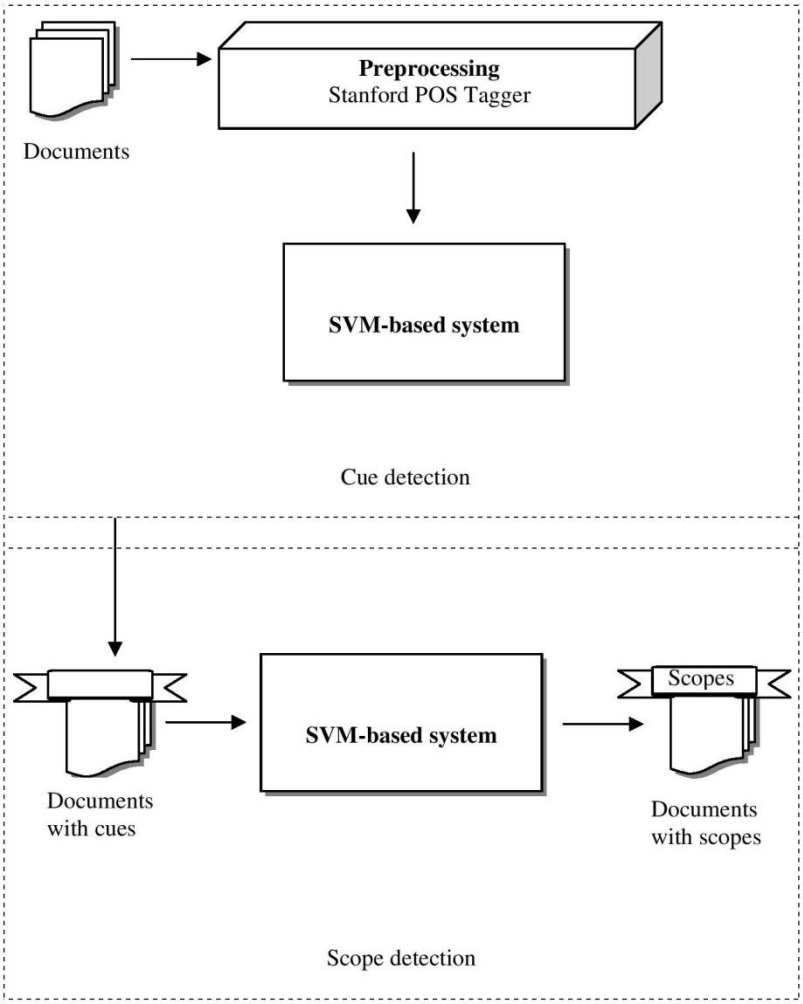

Figure 5.1: System architecture

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

In the first phase, when the cues are detected, a classifier predicts whether each word in a sentence is the first 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 (4), the token *ca* is assigned to the B class; *n't* is tagged as I and the rest of the tokens in the sentence as O class.

(4) Murphy Lee raps about him and how women **ca n't** get enough of him .

In the second step, another classifier decides at sentence level, the words affected by the cues identified in the previous phase. This means determining, for every sentence that have cues, if 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 (4) the classifier tags the words *enough of him* as IS class whereas it assigns the class O to the rest of tokens.

As in the system developed to detect negation and speculation in the biomedical domain, which is described in the Chapter 4, the classifiers are trained using a SVM as implemented in LIBSVM (Chang & Lin, 2011). In addition, the kernel used is the Radial Basic Function (RBF) since this previous experimentation shows its effectiveness in this task. The classifier is parameterised optimising 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 tend toward the majority class. To solve this issue, an algorithmic level solution has been considered, i.e., 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; being 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 like Cao, Zaiane and Zhao (2014) explain, assigning distinct costs to the training examples seems to be the most effective approach of class imbalanced data problem. 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 optimisation (i.e., classes with a higher weight will count more).

It has been also experimented with a Naïve Bayes algorithm implemented in Weka (Witten & Frank, 2005), but as shown in Section 5.3, it produces lower results.

The results of the experiments are detailed in Section 5.3.

5.2.2 Features

All tokens that appear in the collection of documents used for the experimentation are represented by a set of features which 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), i.e., *lemma* and part-of-speech (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 has been followed. It consists of adding a specialised 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 like 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 5.4.

Feature selection experiments (information gained implemented in Weka) 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*.

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

Part-of-speech tags are returned by the Stanford POS tagger[7]

Table 5.4: Features in the cue detection phase

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 5.5 shows the features which directly relate to the characteristics of cues or tokens and their context used in this phase.

Besides the feature set listed in Table 5.5, syntactic features between the token in focus and cues are explored in the classifier, since 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 (2007) note that structure information stored in parse trees helps to identify the scope of negative hedge cues. Both constituent and dependency syntactic

[7] See *http://nlp.stanford.edu/software/tagger.shtml*

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 Ballesteros (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). It has been opted for dependency representations because they are more compact than constituent structures since the number of nodes is constrained on 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.

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)

Part-of-speech tags are returned by the Stanford POS tagger[2]

Table 5.5: Features in the scope detection phase

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; Zhu, Zou, Zhou, 2013), the following (Table 5.6) is the proposal for an operational set of syntactic features.

Feature name	Description
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

Table 5.6: Dependency syntactic features in the scope detection phase

Figure 5.2. Illustration of the corresponding dependency tree of the sentence "The Xterra is no exception."

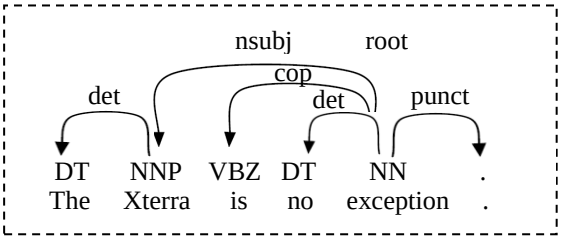

Figure 5.2: Example dependency graph

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 firstly introduced by Gildea and Jurafsky (2002) has been employed as a feature. This feature 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 described in Figure 5.2, (5) shows the dependency graph path between *the* (token in focus) and *no* (cue).

(5) *det* ↑ *nsubj* ↓ *det*

Finally, feature selection experiments (information gained implemented in Weka) 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*.

5.2.3 Post-processing rules

In the cue detection phase, a post-processing algorithm has been 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 post-processing 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 5.3 shows the pseudo code of this algorithm.

This post-processing 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.

```
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
```

Figure 5.3: Cue detection post-processing algorithm pseudo code

5.3 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 has been used, randomly selecting as well as balancing the number of documents in each of these folds.

As detailed in Section 5.2.1, experiments were undertaken with Naïve Bayes and SVM classifiers. Simple baselines models have been also used in both phases to compare the results. The following sections detail the results for the cue and scope detection tasks.

5.3.1 Evaluation and measures

The standard measures employed to assess the performance of the system are the same as those used in the biomedical domain and described in the Chapter 4 (see Section 4.3.1), i.e., precision (P), recall (R) and their harmonic mean F_1-score. However, although F1-score is very popular and suitable for dealing with the class imbalanced 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 on overall performance (Cao et al., 2014), and has been employed by several researchers for evaluating classifiers on imbalanced datasets (Akbani, Kwek, & Japkowicz, 2004; Barua, Islam, Yao, & Murase, 2014).

G-mean is calculated as $\sqrt{}$ 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, a more relaxed approach to measure the percentage of correct scopes is also used. This is put forward by Councill et al. (2010) and it is calculated as the number of correct spans divided by the number of true spans (percentage of correct relaxed scopes, from now on, PCRS). Therefore, in this case, a scope is correct simply if the tokens in the scope have been correctly classified as inside of it.

Finally, a two-tailed sign test applied to the token-level predictions has been employed 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 α=0.05 has been assumed.

5.3.2 Cue detection results

Table 5.7 shows the results for negation and speculation cue detection.

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

Baseline results are shown in the third row of Table 5.7. 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 which cover more than 50% of the total number of cues). In order 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, since 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 post-processing). 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=0.0009). 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.

		Negation				**Speculation**			
	Model	Prec	Rec	**F_1**	**G-mean**	Prec	Rec	**F_1**	**G-mean**
	Baseline	93.54	55.08	69.34	74.20	91.54	57.00	70.26	75.46
Stratified	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

Abbreviations are as follows: 'SVM' = Support Vector Machine; 'RBF' = Radial Basic Function kernel; 'Prec'= Precision; 'Rec' = Recall; 'CS' = Cost-Sensitive Learning.
In brackets, results obtained after applying the post-processing algorithm.
Results obtained by CS-SVM after applying the post-processing algorithm are not shown because they are the same as without applying it. The same occurs with all the speculation detection approachess.
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.

Table 5.7: 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 and F_1 and G-mean (%).

The best F_1 and G-mean for both negation and speculation, as the fifth and seventh rows show, 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

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

In speculation, the results obtained by SVM classifier represents a substantial improvement over the baseline (up by roughly 22%) in terms of F_1 and 10% according to G-mean (see Figure 5.4). It also outstrips the Naïve Bayes results by 20%. As shown by the two-tailed sign test, these differences (p=9.33E-17 compared to the baseline; p=1.69E-14 if it is compared to Naïve Bayes) are significant. The inter-annotator agreement rates may offer some further perspective on the results discussed here. Note that 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, for its part, 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 post-processing, 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 post-processing algorithm (p=0.0013). 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 value of 4.47E-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 5.4. The differences are also significant (p=1.33E-14). In addition, looking at the SFU Review corpus inter-annotator agreement rates for negation cues (F_1 of 92.79% and Kappa value of 92.7%) it could be observed that the results are close to those obtained by a human rater performing the same task.

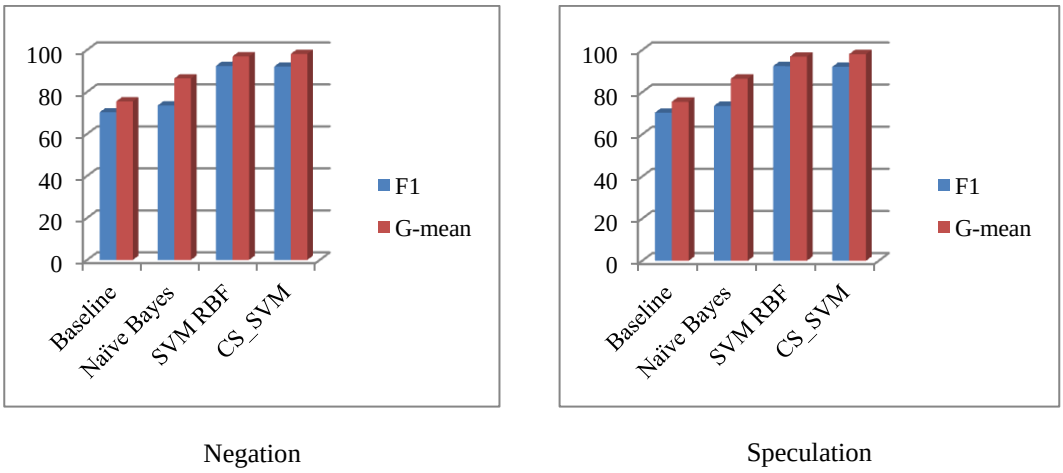

Figure 5.4: Comparison of the results obtained by the different approaches in the cue detection task in terms of F_1 and G-mean (%).

Finally, note 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.

5.3.3 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 has been used. Next, to measure the performance of the whole system, the best scope detection approach has been assessed using the cues identified by the classifier in the previous phase.

Tables 5.8, 5.9 and 5.10 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 recognising the cues. In addition, in contrary to cue detection, the results for speculation are lower than those obtained by negation. It can be explained by the fact that speculation leads to a text with 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 5.2 and 5.3).

Different sets of features have been 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 Part-of-Speech (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 have been added. The last configuration also includes the set of syntactic attributes described in Table 5.6.

In addition, the results are compared with a baseline. It has been proposed as a result of the analysis carried out by Hogenboom, van Iterson, Heerschop, Frasincar and Kaymak (2011) on a set of English movie review sentences. In this 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 non-existent (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 has been 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 5.8, 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 5.9 and 5.10). In the case of speculation, the result is even higher than the

performance obtained by Naïve Bayes using the contextual set of features (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 last columns of Table 5.8 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.

	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

Abbreviations are as follows: '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 5.8: 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 (%).

As explained in Section 5.2.1, Naïve Bayes is not the most suitable classifier to solve 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 5.9). 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.

		Random						Stratified					
	Configuration (features)	Prec	Rec	**F_1**	**G-M**	**PCS**	**PCRS**	Prec	Rec	**F_1**	**G-M**	**PCS**	**PCRS**
	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
Negation	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**
	Configuration (features)	Prec	Rec	**F_1**	**G-M**	**PCS**	**PCRS**	Prec	Rec	**F_1**	**G-M**	**PCS**	**PCRS**
	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
Speculation	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**

Abbreviations are as follows: '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 5.9: 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 (%)

The classifier that best fits the data is SVM. The best results, as Table 5.10 shows, are obtained by adding syntactic 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 reveal the two-tailed sign test (in negation, p values of 0.56, 0.55, 0.50 and 0.35 for F_1, G-mean, PCS and PCRS respectively; in speculation, p=0.54 for F_1, p=0.56 for G-mean, p=0.68 for PCS and p=0.10 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>0.05 in all cases).

		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**
	Configuration (features)	Prec	Rec	**F_1**	**G-M**	**PCS**	**PCRS**	Prec	Rec	**F_1**	**G-M**	**PCS**	**PCRS**
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**

Same notes as in Table 5.9 apply. 'CS' = Cost-Sensitive Learning.

Optimised values of the parameters c and g: c = 32; g= 0.03125

Table 5.10: 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 (%).

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

($p<0.05$ 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=3.06E-17) and the Naïve Bayes classifier (p=2.82E-10) as Figure 5.5 shows. Derived from the figure, it can also be observed considerable differences between CS-SVM and the other approaches in terms of PCRS and F_1.

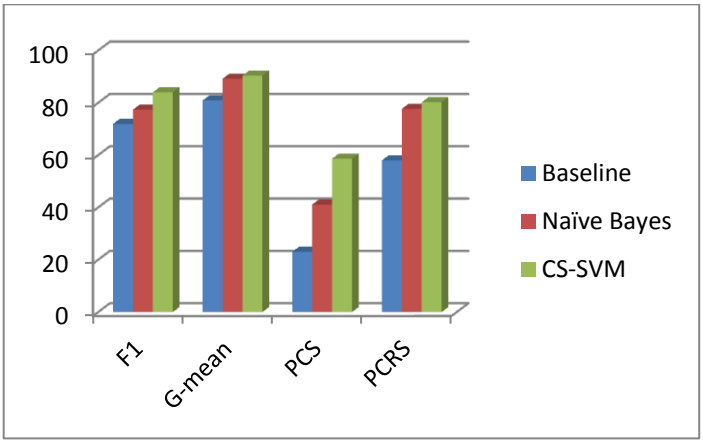

Figure 5.5: Comparison of the results obtained by the different approaches in the negation scope detection task in terms of F_1, G-mean, PCS and PCRS (%).

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 on. 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=1.58E-12 compared to the baseline; p=2.46E-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 can be observed graphically in Figure 5.6.

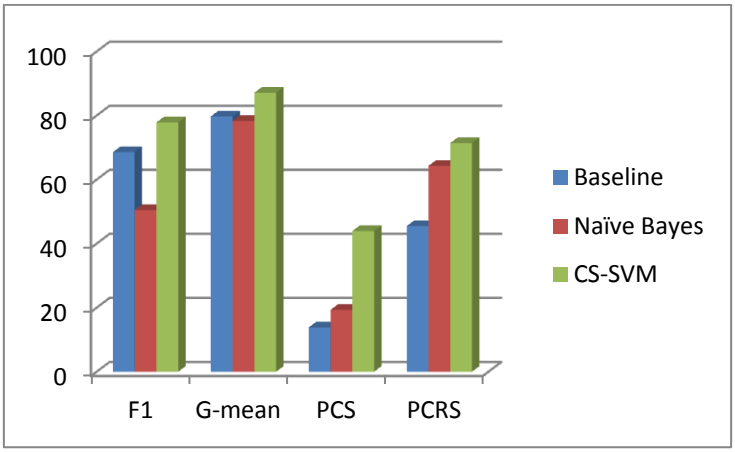

Figure 5.6: Comparison of the results obtained by the different approaches in the speculation scope detection task in terms of F_1, G-mean, PCS and PCRS (%).

Inter-annotator 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 5.11 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 without applying the post-processing step. To identify the scope, the CS-SVM classifier with contextual and dependency syntactic features has been 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 regards 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 with those obtained by an annotator doing the same task. This suggests that when a cue is correctly predicted, its scope is also properly identified.

	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**

Same notes as in Table 5.10 apply

Table 5.11: Results for detecting negation and speculation scopes with predicted cues: Averaged 10-fold cross-validation results for 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 (%).

The results are promising and the system is portable. They are higher than the baseline results, especially in terms 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 great differences are 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 recognising 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 which 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 what makes direct comparison is 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 5.12, 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 shown 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.

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

Table 5.12: 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 (%).

5.3.4 Error analysis

5.3.4.1 Cue detection

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 has been done on the SVM approach (using the random cross-validation for speculation and the stratified one for negation, applying in this last case post-processing), which is the system that achieves the best results. The errors are summarised in Table 5.13 and are mainly due to the ambiguity that characterises 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 post-processing algorithm). In

41 cases (17.15%) for negation and 121 (38.05%) for speculation, errors are words which 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 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. Last type of error is caused by cues with low frequencies of occurrence in the corpus. Examining more closely at the distribution of these words, it can be seen that they appear only once and are due to annotation errors which 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 (22.95%) in speculation.

	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*' stands for multiword cue.

Table 5.13: Errors in the cue detection phase

In the FP errors, the system recognises as cues words that do not appear annotated as such in the corpus. The greatest number of them arises because the system indentifies as cues some words that appear in the corpus mostly classified as such (446 cases in speculation and 570 in negation). On the other hand, 75 times (10.71%) in negation and 23 (4.47%) in speculation, the system only identifies part of an MWC. In negation, all of these cases are

corrected by the post-processing algorithm. In speculation, this cannot be resolved by the post-processing algorithm since almost all the MWC consist of more than 2 words. Finally, another type of error is introduced when the classifier identifies a word as a negation/speculation cue when it have 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).

5.3.4.2 Scope detection

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), since it is the approach that achieves the best results. The most frequent errors are shown in Figure 5.4 and described below; showing examples which compare the correct scope annotation for a cue (Gold Standard, henceforth, GS) with the prediction made by the system (System Detection, hereafter, SD):

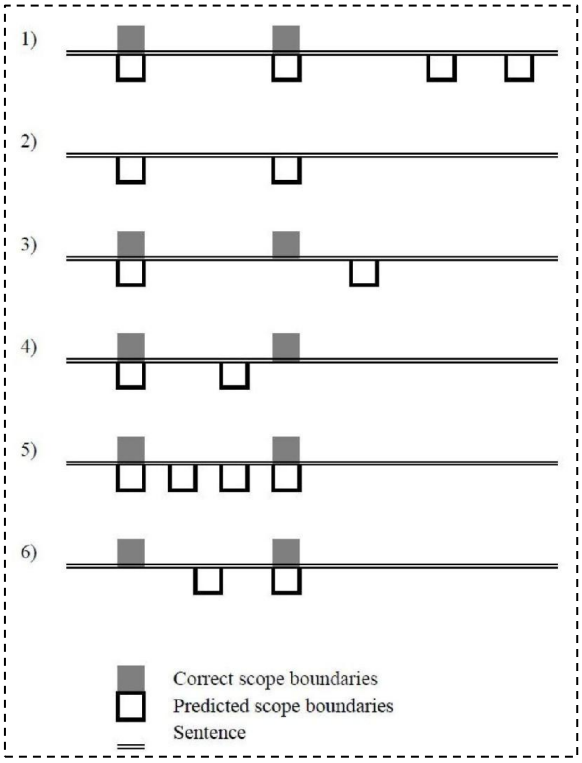

Figure 5.7: Errors in the scope detection phase

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 Section 5.1.2, 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 by extending 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-room [**could have been either a lite-on]** or **[a Samsung**]. (GS)
   The DVD-room [**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 [**recognised either cards**]. (GS)

   The computer never [**recognised**] 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 proportion is higher (17.54%).

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

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

5.4 Conclusions and chapter summary

This chapter 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).

The SFU Review corpus is described in Section 5.1. Specifically, Section 5.1.1 details their annotation process while the main characteristics of the corpus are shown in Section 5.1.2.

The methodology followed to solve the task is presented in Section 5.2. First, the system architecture is described in Section 5.2.1. Basically, the resulting system works in two steps:

in the first one, negation/speculation cues are identified and in the second phase, the full scope of these cues is determined. Next, the set of features used to represent all the tokens that appear in the documents is explained in Section 5.2.2. These attributes are different in each of the two phases into which the problem is divided. Finally, the results yielded by the system in the cue detection task are improved by applying a post-processing algorithm to the output of the classifier whose pseudo code is shown in Section 5.2.3.

The developed system differs from the previous approaches in different aspects. The system architecture is simple and SVM has been employed as classifier algorithm. Although it has proven to be very powerful in text classification task, as far as we concerned, it has hardly been used by other authors to solve this task. Radial Basic Function kernel has been chosen and its parameters have been optimised.

On the other hand, this is a classification problem of imbalanced data sets in which the classification algorithms tend toward the majority class. To solve this issue, an algorithmic level solution has been considered, i.e., Cost Sensitive Learning (CSL) showing that this is an efficient way to address the problem. Evaluation measures suitable for this kind of problems have also been introduced.

Finally, a complete set of features has been employed. They include token-level attributes, contextual as well as syntactic features. In addition, new attributes have been explored (e.g., place of the cue in the sentence).

The results are reported in Section 5.3. In particular, the set of measures employed to assess the performance of the approach is described in Section 5.3.1 while Sections 5.3.2 and 5.3.3 present the results for the cue and scope detection tasks, respectively. Overall, the results show how the proposed method outstrips the baseline by as much as about 20% in the negation cue detection and about 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 achieved 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%). In addition, they 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. Finally, a detailed error analysis is also provided in Section 5.3.4; Section 5.3.4.1 discusses the mistakes introduced by the classifier in the cue detection phase while Section 5.3.4.2 shows the most common errors that appear in the scope recognition phase.

Chapter 6

Conclusions and future work

6.1 Main contributions

This thesis tackles negation and speculation treatment in computational linguistics in the two fields which have received more attention: biomedical and review.

In the biomedical domain, a machine-learning system that identifies the negation/speculation cues and their scope in clinical texts has been developed, using the clinical sub-collection of the BioScope corpus as a learning source and for evaluation purposes. The work is focused on clinical documents because this contribution is part of the project described in de Buenaga et al. (2010). For this reason, the proposed approach may not be generalisable to other domains because the expectations in terms of effectiveness could be different if it was used in a corpus with other features, such as scientific texts. The proposed approach achieves an F_1 of 97.3% and 94.9% in negation and speculation cue detection, respectively. In the scope recognition, the system reports F_1 values of 90.9% in negation and 71.9% in speculation. These results show the superiority of the machine-learning-based approach regarding the use of regular expressions. In fact, in the detection of negation expressions, the developed system outstrips the F_1 of NegEx (Chapman et al., 2001) by 30%. In speculation, the proposed method beats the F_1 of the best system by almost 10%. In addition, compared to other approaches based on machine-learning techniques, the developed global system correctly determines approximately 20% more than the scopes identified by Morante and Daelemans (2009b) in negation. In speculation, this difference is greater and the proposed approach correctly recognises nearly twice the number of scopes identified by Morante and Daelemans (2009a). This means improving the results to date for the sub-collection of clinical documents. However, much still remains to be done since scope

detector performance is far from having reached the level of well established tasks such as *parsing*, especially in speculation detection.

Also in the biomedical field, this thesis includes a comprehensive overview study of tokenization tools. Choosing the right tokenizer in this domain is a non-trivial task so this contribution aims to provide a valuable guideline for NLP developers in the biomedical field to select the appropriate tokenizer as the first phase of a text mining task. Specifically, all the biomedical domain difficulties, together with what it is considered to be the correct tokenization in each of these difficult cases are detailed. The process followed to create the list of tools for tokenizing texts to analyse is also explained, including a description of the technical, functional and usability criteria employed to asses each of these tokenizers. After analyzing 21 tools according to the criteria, 13 of them are tested on a set of 28 sentences from the BioScope corpus. Finally, the two tokenizers that show better features and more accuracy and consistency in the examples tested in the previous phase are evaluated in a subset of sentences of this corpus. This contribution means, as far as we are aware, the first comparative evaluation carried out on tokenizers in the biomedical field.

In the review domain, although negation and speculation recognition can help to improve the effectiveness of sentiment analysis and opinion mining tasks, there is just a few works on detecting negative information. Besides, there is, as far as we are aware, no work in identifying speculation. Therefore, this thesis aims to fill this gap through the development of a system which automatically identifies both negation and speculation keywords and their scope. It means the first attempt to detect speculation in the review domain. The novelty of this contribution also lies in the fact that, to the best of our knowledge, this is the first system trained and tested on the SFU Review corpus (Konstantinova et al., 2012). This corpus is extensively used in opinion mining and consists of 400 documents annotated with negative and speculative information. Overall, the results are competitive and the system is portable. In fact, the results reported in the cue detection task (92.37% and 89.64% in terms of F1 for negation and speculation, respectively) are encouragingly high. In the case of the speculation, the results are comparable to those obtained by a human annotator doing the same task. In the scope detection task, the results are promising and the system correctly identifies 79.13% full scopes in negation and 69.69% in speculation. The proposed approach outstrips the baseline by as much as about 20% in the negation cue detection and improves it up by

roughly 10% in scope detection. In addition, they 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.

Negation/speculation detection systems presented in this thesis, present original aspects compared to previous works. The architecture used is simple and SVM has been chosen as classifier algorithm since it has proven to be very powerful in text classification task and hardly been used by other authors to solve this task. In addition, different kernels have been tested and its parameters have been optimised.

On the other hand, different strategies have been employed to treat the imbalanced data sets trying to avoid that the classification algorithms tend toward the majority class. Supervised resample techniques have been used showing that applying these techniques to the data help solve the problem and improve the system performance. An algorithmic level solution has been considered, i.e., Cost Sensitive Learning showing that this is also an efficient way of tackling the problem. Evaluation measures suitable for this kind of problems have been introduced.

New features, such as the place of the cue in the sentence, the distance between the cue and the token in focus, etc., have been explored. The final set of attributes includes token-level attributes, contextual as well as syntactic features.

Comparing both domains, they differ in many aspects. It highlights that the percentage of negative and speculative information in the review domain is higher than in the biomedical one. Szarvas et al. (2008) report that the number of negative sentences in the BioScope corpus is about 13% and between 18% and 20% of the sentences correspond to speculation, depending on the type of documents. In the SFU corpus, 18% of the sentences include negative information while the percentage of speculative information is 22.7%. It shows that negation is even more relevant in the review domain as well as the proportion of speculation is also higher because this kind of information is widely used to express opinions in the reviews which leads to the text with greater degree of complexity. In addition, clinical documents are characterised for consisting of short sentences, written in a medical language. In reviews, sentence length is much longer than in the clinical data and the style of the text is

more literary, therefore allowing for a greater degree of linguistic richness. This latter also often includes misspelling mistakes.

These differences make the negation/speculation detection difficult and cause that the results yielded by the system developed for the review domain are lower than those obtained by the proposed approach for the clinical domain. In fact, the system performance drops 4% and 2.5% in the negation and speculation cue detection task, respectively (F_1 values of 93.7% in negation and 94.9% in speculation for the clinical domain; 89.6% in negation and 92.3% in speculation for the review domain). In the scope detection, the results in the review domain fall by 9% in negation and 2% in speculation (both in terms of F_1), compared to the clinical domain. Finally, the system developed by the clinical domain correctly identifies about 25% more full scopes than those recognised by the approach proposed for the review domain.

6.2 Future work

Briefly, the main contributions presented in this thesis are focused on the development of machine-learning systems to detect negation, speculation and their linguistic scope in clinical texts as well as in the review domain.

In the clinical domain, the results are high, especially for negation. Future research will be aimed at measuring the robustness of the system when different types of texts from the same domain are applied. To this end, the system will be tested in the paper and abstract sub-collections of the Bioscope corpus. Due to the fact that scientific literature presents more ambiguity and complex expressions than the clinical texts, the proposed approach will have to be adapted in order to fit the needs of the scientific texts. This will be carried out in two aspects. First, in the cue detection phase, external sources of information will be used. They could include drawing information from other parsers and generating cue features from external lexicons. Next, in the scope detection phase, it would be necessary to explore new features derived from deeper syntactic analysis because this information has proven to be effective in the negation and speculation detection system developed for the review field.

In addition, we plan to integrate negation/speculation detection in a clinical record retrieval system. An initial work in this regard can be found in Cordoba et al. (2011).

Overall, the results in the review domain are promising. However, scope detection results can be subject to improvement. Therefore, future research directions include enhancing the performance of the system in this case. Normally, the scope includes whole chunks, i.e., sequence of words that forms a syntactic group. Figure 6.1 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 post-processing 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.

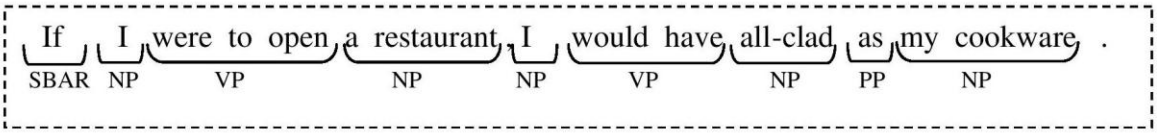

Figure 6.1: Example of shallow parsing

Additionally, it is worth investigating whether correct annotation of negation/speculation improves the results of the SO-CAL system (Taboada, Voll, & Brooke, 2008; Taboada, Brooke, Tofiloski, Voll, & Stede, 2011) using the system described here as a recogniser for this kind of information, rather than the search heuristics that the SO-CAL system is currently using. Thus, as proposed by authors like Councill et al. (2010), it could also be useful to measure the practical impact of accurate negation/speculation detection to check whether it helps to improve the performance in sentiment analysis.

Finally, we plan to study how tokenization results affect the performance of some major text mining applications such as information retrieval, information extraction, text categorization, named entity recognition, drug/drug interaction, negation and speculation detection, etc. It will carry out creating a gold standard for each task and conducting a comparative study based on the performance of these tasks using tokenization results of different tokenizers. This kind of comparative study would empirically justify the strength of the best tokenizers.

Chapter 7

Conclusiones y trabajo futuro

7.1 Principales aportaciones

Esta tesis aborda el tratamiento de la negación y la especulación en lingüística computacional en los dos campos que han recibido más atención: biomédico y artículos de opinión.

En el dominio biomédico, se ha desarrollado un sistema de aprendizaje automático que identifica las palabras claves de negación/especulación así como su alcance, usando la sub-colección de documentos clínicos de la colección BioScope como fuente de aprendizaje así como para fines de evaluación. El trabajo está centrado en documentación clínica porque esta contribución es parte del proyecto descrito en de Buenaga et al. (2010). El enfoque propuesto obtiene una medida F_1 de 97,3% y 94,9% en la detección de las palabras clave de negación y especulación, respectivamente. En el reconocimiento del alcance, el sistema presenta valores F_1 de 90,9% en negación y 71,9% en especulación. Estos resultados muestran la superioridad del enfoque basado en aprendizaje automático respecto del uso de expresiones regulares. De hecho, en la detección de las expresiones de negación, el sistema desarrollado supera la medida F_1 obtenida por NegEx (Chapman et al., 2001) en un 30%. En especulación, el método propuesto mejora la medida F_1 del mejor sistema en casi un 10%. Además, comparado con otros enfoques basados en técnicas de aprendizaje automático, el sistema global desarrollado determina correctamente aproximadamente un 20% más de los alcances identificados por Morante y Daelemans (2009b) en negación. En especulación, esta diferencia es mayor y el enfoque propuesto correctamente reconoce casi dos veces más alcances de los detectados por Morante y Daelemans (2009a). Esto significa mejorar los resultados hasta la fecha para la sub-colección de documentos clínicos. Sin embargo, aún hay mucho por hacer ya que el rendimiento de los reconocedores del alcance está lejos de haber alcanzado los niveles de

tareas tradicionales como el *análisis sintáctico*, especialmente en la detección de la especulación.

También en el campo biomédico, esta tesis incluye un completo estudio general de herramientas de tokenización. Elegir el tokenizador correcto en este ámbito no es una tarea trivial por lo que esta contribución tiene como objetivo proporcionar a los desarrolladores de Procesamiento del Lenguaje Natural en el campo biomédico, una guía valiosa para seleccionar el tokenizador apropiado como primera fase de una tarea de minería de texto. En concreto, se detallan todas las dificultades ámbito biomédico, junto con lo que se considera que es la tokenización correcta en cada uno de estos casos difíciles. También se explica el proceso seguido para crear la lista de herramientas para tokenizar textos a analizar, incluyendo una descripción de los criterios técnicos, funcionales y de usabilidad empleados para evaluar cada una de estos tokenizadores. Tras el análisis de 21 herramientas de acuerdo con los criterios, 13 de ellas han sido probadas en una serie de 28 frases de la colección de documentos Bioscope. Por último, los dos tokenizadores que muestran mejores características y más precisión y coherencia en los ejemplos analizados en la fase anterior, se evalúan en un subconjunto de frases de esta colección. Esta contribución supone, hasta donde alcanza nuestro conocimiento, la primera evaluación comparativa llevada a cabo en tokenizadores en el campo biomédico.

En el dominio de artículos de opinión, aunque el reconocimiento de la negación y la especulación pueden ayudar a mejorar la eficacia de las tareas de análisis de sentimiento y minería de opinión, hay sólo unos pocos trabajos sobre la detección de la información negativa. Además, no hay, hasta donde sabemos, ningún trabajo en la identificación de la especulación. Por lo tanto, esta tesis pretende corregir esta deficiencia a través del desarrollo de un sistema que reconoce automáticamente tanto las palabras clave de negación y especulación como su alcance. Esto supone el primer intento en detectar la especulación en el dominio de artículos de opinión. La novedad de esta aportación reside también en el hecho de que, según nuestro conocimiento, éste es el primer sistema entrenado y evaluado en el SFU Review corpus (Konstantinova et al., 2012). Este corpus es ampliamente utilizado en la minería de opinión y consiste en 400 documentos anotados con información negativa y especulativa. En general, los resultados son competitivos y el sistema es portable. De hecho, los resultados mostrados en la tarea de la detección de las palabras clave (92,37% y 89,64%

en términos de F_1 para la negación y la especulación, respectivamente) son alentadoramente altos. En el caso de la especulación, los resultados son comparables a los obtenidos por un anotador humano realizando la misma tarea. En detección del alcance, los resultados son prometedores y el sistema identifica correctamente 79,13% de los alcances completos en la negación y 69,69% en la especulación. El enfoque propuesto supera el algoritmo de línea de base en aproximadamente un 20% en la detección de las palabras clave negativas y lo mejora en un 10% aproximadamente en la detección de alcance. Además, se muestra que, de acuerdo con los comentarios de otros autores, la información léxica es suficiente para identificar de forma automática las palabras clave, mientras que, para determinar eficazmente el alcance de una palabra clave, es necesario incluir atributos sintácticos.

Los sistemas de detección de la negación/especulación detallados en esta tesis presentan aspectos originals respecto a los trabajos previos. Se usa una arquitectura simple en la que SVM ha sido elegido como algoritmo de clasificación ya que se ha demostrado que es muy potente en tareas de clasificación de texto a la vez que apenas ha sido empleado por otros autores para resolver esta tarea. Se han probado diferentes kernels y sus parámetros se han optimizado.

Por otro lado, se han utilizado diferentes estrategias para el tratamiento de datos no balanceados a fin de evitar que los algoritmos de clasificación tiendan hacia la clase mayoritaria. En conreto, se han usado técnicas de resample, mostrando que aplicar este tipo de técnicas sobre los datos ayuda a solucionar el problema y mejora el rendimiento del sistema. Además, se ha considerado una solución algorítmica (Aprendizaje Sensitivo al Costo), mostrando que ésta es también una forma eficiente de abordar el problema. Medidas de evaluación adecuadas para este tipo de problemas se han introducido.

Se han explorado nuevos atributos, como el lugar que ocupa la particular en la frase, la distancia entre la particula y la palabra bajo análisis, etc. El conjunto final de atributos incluye características a nivel de token, contexto así como atributos sintácticos.

Comparando ambos dominios, éstos difieren en varios aspectos. Destaca que el porcentaje de información negativa y especulativa en el dominio de artículos de opinión es mayor que en el biomédico. Szarvas et al. (2008) indica que el número de frases negativas en la colección de

documentos BioScope es alrededor del 13% y que entre el 18% y el 20% de las frases corresponden con la especulación, dependiendo del tipo de documentos. En la colección de documentos SFU, el 18% de las frases incluyen información negativa, mientras que el porcentaje de información especulativa es del 22,7%. Esto muestra que la negación es aún más relevante en el ámbito de artículos de opinión, así como que la especulación es también mayor debido a que este tipo de información es ampliamente utilizada para expresar opiniones, dotando al texto de un mayor grado de complejidad. Además, los documentos clínicos se caracterizan por consistir en frases cortas, escritas en lenguaje médico. En los artículos de opinión, las frases son mucho más largas que en los datos clínicos y el estilo del texto es más literario, lo que permite por tanto, un mayor grado de riqueza lingüística. Este último también incluye a menudo errores ortográficos.

Estas diferencias dificultan la detección de la negación/especulación y provocan que los resultados obtenidos por el sistema desarrollado para el dominio de artículos de opinión sean inferiores a los obtenidos por el método propuesto para el dominio clínico. De hecho, el rendimiento del sistema cae 4% y 2,5% en la tarea de la detección de la palabra clave negativa y especulativa, respectivamente (valores de F_1 de 93,7% en negación y 94,9% en especulación para el dominio clínico; 89,6% en negación y 92,3% en especulación en el dominio de artículos de opinión). En la detección del alcance, los resultados del dominio de artículos de opinión bajan un 9% en negación y el 2% en especulación (ambos en términos de F_1), en comparación con el dominio clínico. Por último, el sistema desarrollado para el dominio clínico identifica correctamente aproximadamente un 25% más de alcances completos que los reconocidos por el enfoque propuesto para el dominio de artículos de opinión.

7.2 Trabajo futuro

Brevemente, las principales aportaciones presentadas en esta tesis se centran en el desarrollo de sistemas de aprendizaje automático para detectar la negación, la especulación y su ámbito lingüístico en textos clínicos, así como en artículos de opinión.

En el ámbito clínico, los resultados son altos, especialmente para la negación. La investigación futura se orientará a medir la robustez del sistema cuando se aplican diferentes tipos de textos del mismo dominio. Para ello, el sistema será evaluado en las sub-colecciones

de artículos científicos y resúmenes de artículos científicos de la colección de documentos BioScope. Debido a que la literatura científica presenta más ambigüedad y expresiones complejas que los textos clínicos, el enfoque propuesto tendrá que ser adaptado con el fin de satisfacer las necesidades de los textos científicos. Esto se llevará a cabo en dos aspectos. En primer lugar, en la fase de detección de las palabras clave, se utilizarán fuentes de información externas. Éstas podrían incluir información derivada de otros analizadores sintácticos así como la generación de atributos de la palabra clave a partir de diccionarios externos. A continuación, en la fase de detección de alcance, sería necesario explorar nuevas características derivadas de un análisis sintáctico más profundo ya que esta información ha demostrado ser eficaz en el sistema de detección de la negación y la especulación desarrollado para el campo de artículos de opinión.

Además, está previsto integrar la detección de la negación/especulación en un sistema de recuperación de historiales clínicos. Un trabajo previo en este sentido puede encontrarse en Cordoba et al. (2011).

En general, los resultados en el dominio de artículos de opinión son prometedores. No obstante, los resultados de la detección de alcance pueden ser objeto de mejora. Por lo tanto, las futuras líneas de investigación incluyen la mejora del rendimiento del sistema en este caso. Normalmente, el alcance incluye sintagmas completos, es decir, secuencias de palabras que forman un grupo sintáctico. La Figura 7.1 muestra un ejemplo donde la palabra clave es *if* y el alcance consiste en los sintagmas *were to open* y *a restaurant*. *Shallow processing (chunking)* aplicado en la fase de post-procesamiento podría ayudar a corregir los límites de alcance identificados por el clasificador en los casos en los que no se incluyan grupos sintácticos de palabras completos.

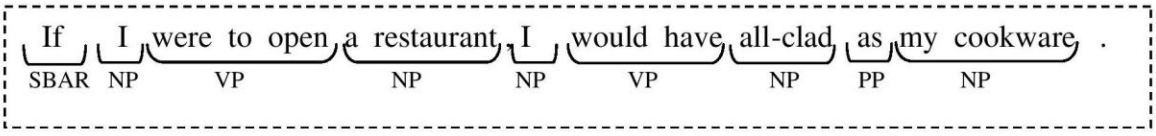

Figura 7.1: Ejemplo de shallow parsing

Adicionalmente, merece la pena investigar si la correcta anotación de la negación/especulación mejora los resultados del sistema SO-CAL (Taboada et al., 2008; Taboada et al., 2011) usando el sistema aquí descrito como reconocedor de este tipo de

información, en lugar de la heurística de búsqueda que el sistema SO-CAL utiliza actualmente. Por lo tanto, según lo propuesto por autores como Councill et al. (2010), podría ser también útil medir el impacto práctico de la correcta detección de la negación y especulación para comprobar si ésta mejora el rendimiento del análisis de sentimientos.

Finalmente, está previsto estudiar cómo los resultados de la tokenización afectan al rendimiento de las principales aplicaciones de minería de texto, tales como la recuperación de información, extracción de información, categorización de texto, reconocimiento de entidades nombradas, interacción de fármacos, detección de la negación y la especulación, etc. Esto se llevará a cabo creando un gold standard para cada tarea y realizando un estudio comparativo basado en el rendimiento de cada una de estas tareas utilizando los resultados de tokenización de los distintos tokenizadores. Este tipo de estudio comparativo justificará empiricamente el potencial de las mejores herramientas de tokenización.

Appendix A

Description of the tokenization tools analysed in the tokenization problem

This appendix describes in detail each of the tokenizers analysed in the study of the tokenization problem in the biomedical domain. Tools are presented in alphabetical order.

Brill's POS tagger

Brill's POS tagger (Brill, 1992) is designed by Eric Brill. It can be summarised as an *error-driven transformation-based tagger*: Error-driven in the sense that it recourses to a supervised learning and transformation-based because a tag is assigned to each word and changed using a set of predefined rules. It achieves quite a high accuracy. There is a Java implementation developed by Jimmy Lin. It is an open resource tool and features a simple installation. However, it does not work well as a tokenizer since it removes some tokens and performs poorly for biomedical words such as *substance names*.

Dan Melamed's tokenizer

Dan Melamed's tokenizer is one of the 170 general text processing tools developed by him. It is written in Perl and its source code is available. However, there is no documentation, support or even information about how the tokenizer works.

English Resource Grammar

The English Resource Grammar (ERG) (Copestake & Flickinger, 2000; Flickinger, 2000) is a broad-coverage, linguistically precise HPSG-based grammar of English developed by Dan Flickinger. It is semantically grounded in Minimal Recursion Semantics. The ERG uses the LKB grammar engineering environment for development, and the relatively efficient PET

parser, among others, for applications. The usability of this tool is not good since its documentation is poor and a lot of effort is necessary to learn how to install it and use it. Therefore, the tokenizer is tested using the online demo.

Freeling

Freeling (Carreras et al., 2004; Padró & Stanilovsky, 2012) is developed at the Center for Language and Speech Technologies and Applications (Technical University of Catalunya). It is an open source language analysis tool suite written in C++. Its documentation is useful, well written and complete. It provides an online demo which is what it has been used to test the tokenization. It is a general purpose tool so maybe for this reason it does not perform very well in the biomedical domain.

Genia tagger

Genia tagger (Kulick et al., 2004; Tsuruoka et al., 2005; Tsuruoka & Tsujii, 2005) is certainly one of the most popular tools in the biomedical domain. It is specifically tuned for biomedical text such as *MEDLINE abstracts*. As many of the tools described in this section it has an online demo so this functionality has been employ to test it. The documentation is poor but yet it is easy to use.

Gate Unicode tokenizer

Gate Unicode tokenizer (Cunningham et al., 2002) is a resource included in ANNIE plug-in that can be used through the GATE framework (Cunningham, Tablan, Roberts, & Bontcheva, 2013), an open source software capable of solving almost any text processing problem. Therefore, the use of this tokenizer might not be intuitive for those not familiar with GATE. The tokenizer consists of a regular tokenizer and Java Annotation Patterns Engine transducer. The latter adapts the generic output of the tokenizer to the requirements of the English POS tagger. Nevertheless, this tool does not show a great accuracy in tokenizing the text.

Gump tokenizer

Gump tokenizer is designed by Torbjörn Lager and it is a natural language tokenizer based on the Gump programming language. Due to this fact it might be an unsuitable tool for users who want to customize the tokenization process. Among other things, it does not need to be

accompanied by a sentence splitter, since it handles sentence splitting all by itself. Although it may be enough for general purpose tokenization, it introduces some errors in cases such as those with hypertext markup symbols which usually appear in the biomedical domain.

JULIE LAB tokenizer

JULIE lab tokenizer (Tomanek et al., 2007b) is a machine learning-based tool, developed and optimised for handling life science documents. However, it introduces some errors, mainly in words with hyphens. It is available as UIMA components and as stand-alone tool. The tokenizer is easy to use and the support is really good.

LingPipe

Lingpipe (Carpenter & Baldwin, 2011) is a software library for NLP implemented in Java. It includes a complete Java API and a demo online, so it is easy test the tool with your own examples. The documentation is fine; however, the support is only available in some of the paid versions.

LT TTT

The LT TTT (Text Tokenization Tool) (Grover et al., 2000) is a toolset developed at the University of Edinburgh. The last version of the software, TTT2, has been used. It provides a flexible means of tokenizing texts and adding linguistic markup at various levels. Although it has some features in treating words that include numbers as well as in treating hyphenated compound words, it fails in many cases.

Mallet tokenizer

MALLET (McCallum, 2002) has been developed at the University of Massachusetts and it is a Java-based package for statistical NLP, document classification, clustering, IE, and other machine learning applications to text. It is hard to learn how to use the tokenizer in some way because the documentation is poor. In addition, its performance is not very good because it removes a great number of tokens from the original text.

McClosky-Charniak parser

McClosky-Charniak parser (McClosky & Charniak, 2008; McClosky & Adviser-Charniak, 2010), also called BLLIP parser, is one of the currently most used parsers in the biomedical

domain. The default parser and re-ranker models are trained on Wall Street Journal data but there are publicly available models which are trained on biomedical text, namely the Genia corpus. Although it does not have any support and the documentation consists just of several "readme" files, it is easy to use. As with the Stanford POS tagger, it converts parentheses and squared brackets into the same normalised tokens (-LRB- or -RRB-).

MedPost

MedPost (Smith et al., 2004) is a POS tagger for MEDLINE developed at the U.S. National Library of Medicine. It is one of the most popular tools designed for the biomedical domain. Although it is a POS tagger, it includes an option to tokenize the text. It is written in Perl and C++ but there is a Java implementation available which has been used in this experimentation. The usage of the tool is very difficult and rather non-intuitive.

MXPOST tagger

MXPOST (Ratnaparkhi, 1996) is developed by Adwait Ratnaparkhi. It is a maximum entropy POS tagger written in Java and designed for the biomedical domain. There is little information about this tool and furthermore, the binaries are not available. Therefore, it has not been possible to experiment with it.

NLTK tokenizer

NLTK (Natural Language Toolkit) (Bird et al., 2009) has been developed at the University of Pennsylvania and it is written in Python. It is a free, open source, community-driven project. It provides easy-to-use interfaces to over 50 corpora and lexical resources such as *WordNet*, along with a suite of text processing libraries for classification, tokenization, stemming, tagging, parsing, and semantic reasoning. There is also a demo online which is the option used to carry out the experimentation. Although the installation is not simple, the documentation is really good, which makes its later use easier.

OpenNLP tokenizer

OpenNLP is a machine learning based toolkit for the processing of natural language text such as *tokenization*, *sentence segmentation*, *POS tagging* and so on. It is written in Java and it includes an API. The main problem of the tokenizer is that it treats parentheses

inconsistently. Nevertheless, it should be highlighted that the grammar model is trainable so the user could train the system with a customized sentence set.

Penn Bio tokenizer

Penn Bio tokenizer (Jin et al., 2006; McDonald & Pereira, 2005; McDonald et al., 2004) is a tokenizer written in Java and focused on the biomedical domain. There is little information about this system but its installation and use are easy. There is a plug-in that can be used through the GATE framework, so the experimentation has been carried out in this way.

Stanford POS tagger

Stanford POS tagger (Toutanova et al., 2003) is a Java implementation of the log-linear POS taggers originally written by Kristina Toutanova. It is developed at the Stanford University. Although the effort needed to learn how the tool works is moderate, its later installation and use is simple. Despite being a general purpose tool, it performs well in the biomedical domain. However, it converts parentheses and squared brackets into the same normalised tokens (-LRB- or -RRB-).

Specialist NLP

Specialist (Browne et al., 2003) is developed at the U.S. National Library of Medicines. It is a set of resources for the analysis of free text documents into words, terms, phrases, sentences and sections. It is written in Java and it is rather popular in the biomedical domain. The tokenizer package can handle free text and MEDLINE citation formats. Although this package is meant to be called from the Java API embedded within other applications, it does include a program that can be called from a command line. It is focused on biomedicine but it does not manage well the hypertext markup symbols, substance names and even decimals or ranges.

UIUC word splitter

UIUC word splitter is developed at the University of Illinois. It is a simple Perl script that reads plain text and outputs the words with spaces between every word and punctuation marks. To work with the tool, it is necessary to pre-process the text with a sentence splitter before using the tokenizer because its input should include one sentence per line. It converts parentheses and squared brackets into the same normalised tokens (-LRB- or -RRB-). It does not include documentation or support but this does not affect its usability.

Xerox tokenizer

Xerox (Beesley & Karttunen, 2003) provides a set of tools used in many linguistic applications such as *morphological analysis*, *tokenization* and *shallow parsing* of a wide variety of natural languages. The finite state tools here are built on top of a software library that provides algorithms to create automata from regular expressions and equivalent formalisms and contains both classical operations, such as *union* and *composition*, and new algorithms such as *replacement* and *local sequentialisation*. The tokenizer could be easily managed through the web interface as well as a SOAP API of the web service to integrate it into applications. This tokenizer is highly suitable for any real-world NLP task.

Appendix B

Set of sentences used to test the tokenization tools

This appendix details the set of 28 sentences from the BioScope corpus used to test the tokenization tools and its correct tokenization.

Normal ^ chest ^ **x-ray** ^ .

2-year ^ **2-month** ^ old ^ female ^ with ^ pneumonia ^ .

This ^ may ^ occur ^ through ^ the ^ ability ^ of ^ **IL10** ^ to ^ induce ^ expression ^ of ^ the ^ gene ^ , ^ suppressor ^ of ^ cytokine ^ cueing ^ 3 ^ **(** ^ SOCS3 ^ **)** ^ .

The ^ results ^ identify ^ functionally ^ distinct ^ epitopes ^ on ^ the ^ CD4 ^ coreceptor ^ involved ^ in ^ activation ^ of ^ the ^ **Ras** ^ / ^ **protein** ^ kinase ^ C ^ and ^ calcium ^ pathways ^ .

The ^ maximal ^ effect ^ is ^ observed ^ at ^ the ^ **IL-10** ^ concentration ^ of ^ 20 ^ **U** ^ / ^ **ml** ^ .

These ^ results ^ indicate ^ that ^ within ^ the ^ **TCR** ^ / ^ **CD3** ^ cue ^ transduction ^ pathway ^ both ^ PKC ^ and ^ calcineurin ^ are ^ required ^ for ^ the ^ effective ^ activation ^ of ^ the ^ IKK ^ complex ^ and ^ **NF-kappaB** ^ in ^ T ^ lymphocytes ^ .

Small ^ , ^ scarred ^ right ^ kidney ^ , ^ below ^ more ^ than ^ 2 ^ standard ^ deviations ^ in ^ size ^ for ^ **patient** ^ **'s** ^ age ^ .

If ^ both ^ the ^ best ^ hits ^ of ^ the ^ **N-** ^ and ^ **C-terminal** ^ parts ^ are ^ statistically ^ significant ^ **(** ^ **Evalue** ^ **&lt ;** ^ **10-5** ^ **)** ^ , ^ and ^ distance ^ between ^ them ^ is ^ less ^ than ^ 1 ^ kbp ^ , ^ we ^ call ^ these ^ hits ^ **'** ^ **syntenic** ^ **hits** ^ ^ .

The ^ false ^ positive ^ rate ^ **(** ^ FPR ^ **)** ^ of ^ our ^ predictor ^ was ^ estimated ^ by ^ the ^ method ^ of ^ **D'Haeseleer** ^ and ^ Church ^ 1855 ^ and ^ used ^ to ^ compare ^ it ^ to ^ other ^ prediction ^ datasets ^ .

Of ^ these ^ , ^ Diap1 ^ has ^ been ^ most ^ extensively ^ characterized ^ ; ^ it ^ can ^ block ^ cell ^ death ^ caused ^ by ^ the ^ ectopic ^ expression ^ of ^ reaper ^ , ^ hid ^ , ^ and ^ grim ^ **(** ^ reviewed ^ in ^ **[** ^ 26 ^ **]** ^ **)** ^ .

These ^ results ^ reveal ^ a ^ central ^ role ^ for ^ **CaMKIV** ^ / ^ **Gr** ^ as ^ a ^ **Ca(2+)-regulated** ^ activator ^ of ^ gene ^ transcription ^ in ^ T ^ lymphocytes ^ .

Two stop codons of an iORF (**i.e.** the inframe and **C-terminal** stops) can be any combination of canonical stop codons (TAA, TAG, TGA).

Selenocysteine and pyrrolysine are the **21st** and **22nd** amino acids, which are genetically encoded by stop codons.

A total of **26,003** iORF satisfied the above criteria.

The patient had prior **x-ray** on **1/2** which demonstrated no pneumonia.

Indeed, it has been estimated recently that the current yeast and human protein interaction maps are only **50%** and **10%** complete, respectively 18.

The dotted line indicates significance level **0.05** after a correction for multiple testing.

E-selectin is induced within **1-2** h, peaks at **4-6** h, and gradually returns to basal level by 24 h.

2.

1. Bioactivation of sulphamethoxazole (SMX) to chemically-reactive metabolites and subsequent protein conjugation is thought to be involved in SMX hypersensitivity.

Mutants in Toll cueing pathway were obtained from **Dr. S.** Govind: cactE8, cactIIIG, and cactD13 mutations in the cact gene on Chromosome II.

The transcripts were detected in all the **CD4- CD8-**, **CD4+ CD8+**, **CD4+ CD8-**, and **CD4 CD8+** cell populations.

Footprinting analysis revealed that the identical sequence **CCGAAACTGAAAAGG**, designated E6, was protected by nuclear extracts from B cells, T cells, or HeLa cells.

Bcd mRNA transcripts of **<** or = **2.6** kb were selectively expressed in PBL and testis of healthy individuals.

Names of all available Trace Databases were taken from a list of databases at **http://www.ncbi.nlm.nih.gov/blast/mmtrace.shtml**

This was last documented on the Nuclearv Cystogram dated **1/2/01**.

We found **IL-2Ralpha** expression to be increased in BAL cells from involved sites of active pulmonary tuberculosis.

Expression of a highly specific protein inhibitor for cyclic **AMP-dependent** protein kinases in **interleukin-1 (IL-1)-responsive** cells blocked **IL-1-induced** gene transcription that was driven by the kappa immunoglobulin enhancer or the human immunodeficiency virus long terminal repeat.

In bold, problematic cases in which tokenizers usually fail.

Appendix C

Output of each tokenization tool in the set of sentences

This appendix includes the outputs from each tokenizer for a set of 28 sentences from the BioScope corpus. It also includes how each tool has been tested: local installation, online demo, application programming interface (API).
The ^ mark indicates token boundaries. The errors of each tokenizer are shown in bold.

NLTK tokenizer. It is tested using the online demo. This tokenizer provides 4 types of tokenization: TreebankWordTokenizer, WordPunctTokenizer, PunctWordTokenizer and WhitespaceTokenizer. The first one has been used since the last three introduced too many errors.

Normal^chest^x-ray^.

2-year^2-month^old^female^with^pneumonia^.

This^may^occur^through^the^ability^of^IL-10^to^induce^expression^of^the^gene^,^ suppressor^of^cytokine^cueing^3^(^SOCS3^)^.

The^results^identify^functionally^distinct^epitopes^on^the^CD4^coreceptor^involved^in^ activation^of^the^**Ras/protein**^kinase^C^and^calcium^pathways^.

The^maximal^effect^is^observed^at^the^IL-10^concentration^of^20^**U/ml**^.

These^results^indicate^that^within^the^**TCR/CD3**^cue^transduction^pathway^both^PKC^and^ca lcineurin^are^required^for^the^effective^activation^of^the^IKK^complex^and^NF-kappaB^in^T^lymphocytes^.

Small^,^scarred^right^kidney^,^below^more^than^2^standard^deviations^in^size^for^patient^'s ^age^.

If both the best hits of the N- and C-terminal parts are statistically significant (E-value **<** 10-5), and distance between them is less than 1 kbp, we call these hits **'syntenic hits'**.

The false positive rate (FPR) of our predictor was estimated by the method of D'Haeseleer and Church 1855 and used to compare it to other prediction datasets.

Of these, Diap1 has been most extensively characterized; it can block cell death caused by the ectopic expression of reaper, hid, and grim (reviewed in [26]).

These results reveal a central role for **CaMKIV/Gr** as a **Ca(2+)-regulated** activator of gene transcription in T lymphocytes.

Two stop codons of an iORF (**i.e.** the inframe and C-terminal stops) can be any combination of canonical stop codons (TAA, TAG, TGA).

Selenocysteine and pyrrolysine are the 21st and 22nd amino acids, which are genetically encoded by stop codons.

A total of 26,003 iORF satisfied the above criteria.

The patient had prior x-ray on 1/2 which demonstrated no pneumonia.

Indeed, it has been estimated recently that the current yeast and human protein interaction maps are only **50**% and **10**% complete, respectively 18.

The dotted line indicates significance level 0.05 after a correction for multiple testing.

E-selectin is induced within 1-2 h, peaks at 4-6 h, and gradually returns to basal level by 24 h.

2.

1. Bioactivation of sulphamethoxazole (SMX) to chemically-reactive metabolites and subsequent protein conjugation is thought to be involved in SMX hypersensitivity.

Mutants in Toll cueing pathway were obtained from Dr. S. Govind: cactE8, cactIIIG, and cactD13 mutations in the cact gene on Chromosome II.

The transcripts were detected in all the CD4- CD8-, CD4+ CD8+, CD4+ CD8-, and CD4- CD8+ cell populations.

Footprinting analysis revealed that the identical sequence CCGAAACTGAAAAGG, designated E6, was protected by nuclear extracts from B cells, T cells, or HeLa cells.

Bcd mRNA transcripts of **<** or = 2.6 kb were selectively expressed in PBL and testis of healthy individuals.

Names of all available Trace Databases were taken from a list of databases at **http://www.ncbi.nlm.nih.gov/blast/mmtrace.shtml**

This was last documented on the Nuclearv Cystogram dated 1/2/01.

We found IL-2Ralpha expression to be increased in BAL cells from involved sites of

active^pulmonary^tuberculosis^.

Expression^of^a^highly^specific^protein^inhibitor^for^cyclic^AMP-dependent^protein^kinases^in^interleukin-1^**(^IL-1^)^-^responsive**^cells^blocked^IL-1-induced^gene^transcription^that^was^driven^by^the^kappa^immunoglobulin^enhancer^or^the^human^immunodeficiency^virus^long^terminal^repeat^.

McClosky-Charniak parser. It has been tested by installing locally.

Normal^chest^x-ray^.

2-year^2-month^old^female^with^pneumonia^.

This^may^occur^through^the^ability^of^IL-10^to^induce^expression^of^the^gene^,^suppressor^of^cytokine^cueing^3^(^SOCS3^)^.

The^results^identify^functionally^distinct^epitopes^on^the^CD4^coreceptor^involved^in^activation^of^the^**Ras/protein**^kinase^C^and^calcium^pathways^.

The^maximal^effect^is^observed^at^the^IL-10^concentration^of^20^**U/ml**^.

These^results^indicate^that^within^the^**TCR/CD3**^cue^transduction^pathway^both^PKC^and^calcineurin^are^required^for^the^effective^activation^of^the^IKK^complex^and^NF-kappaB^in^T^lymphocytes^.

Small^,^scarred^right^kidney^,^below^more^than^2^standard^deviations^in^size^for^patient^'s^age^.

If^both^the^best^hits^of^the^N-^and^C-terminal^parts^are^statistically^significant^(^E-value^**<**^10-5^)^,^and^distance^between^them^is^less^than^1^kbp^,^we^call^these^hits^'^syntenic^hits^'^.

The^false^positive^rate^(^FPR^)^of^our^predictor^was^estimated^by^the^method^of^**D^'^Haeseleer**^and^Church^1855^and^used^to^compare^it^to^other^prediction^datasets^.

Of^these^,^Diap1^has^been^most^extensively^characterized^;^it^can^block^cell^death^caused^by^the^ectopic^expression^of^reaper^,^hid^,^and^grim^(^reviewed^in^**[26^]**^)^.

These^results^reveal^a^central^role^for^**CaMKIV/Gr**^as^a^**Ca^(^2+^)^-regulated**^activator^of^gene^ transcription^in^T^lymphocytes^.

Two^stop^codons^of^an^iORF^(^i.e.^the^inframe^and^C-terminal^stops^)^can^be^any^combination^of^ canonical^stop^codons^(^TAA^,^TAG^,^TGA^)^.

Selenocysteine^and^pyrrolysine^are^the^21st^and^22nd^amino^acids^,^which^are^ genetically^encoded^by^stop^codons^.

A^total^of^26,003^iORF^satisfied^the^above^criteria^.

The^patient^had^prior^x-ray^on^1/2^which^demonstrated^no^pneumonia^.

Indeed^,^it^has^been^estimated^recently^that^the^current^yeast^and^human^protein^interaction^maps^are^only^**50^**%^and^**10^**%^complete^,^respectively^18^.

The dotted line indicates significance level 0.05 after a correction for multiple testing.

E-selectin is induced within **1-2h**, peaks at **4-6h**, and gradually returns to basal level by **24h**.

2.

1. Bioactivation of sulphamethoxazole (SMX) to chemically-reactive metabolites and subsequent protein conjugation is thought to be involved in SMX hypersensitivity.

Mutants in Toll cueing pathway were obtained from Dr. S. Govind: cactE8, cactIIIG, and cactD13 mutations in the cact gene on Chromosome II.

The transcripts were detected in all the CD4- CD8-, CD4+ CD8+, CD4+ CD8-, and CD4- CD8+ cell populations.

Footprinting analysis revealed that the identical sequence CCGAAACTGAAAAGG, designated E6, was protected by nuclear extracts from B cells, T cells, or HeLa cells.

Bcd mRNA transcripts of **<** or = **2.6kb** were selectively expressed in PBL and testis of healthy individuals.

Names of all available Trace Databases were taken from a list of databases at **http://www.ncbi.nlm.nih.gov/blast/mmtrace.shtml**

This was last documented on the Nuclearv Cystogram dated 1/2/01.

We found IL-2Ralpha expression to be increased in BAL cells from involved sites of active pulmonary tuberculosis.

Expression of a highly specific protein inhibitor for cyclic AMP-dependent protein kinases in interleukin-1 **(IL-1)-responsive** cells blocked IL-1-induced gene transcription that was driven by the kappa immunoglobulin enhancer or the human immunodeficiency virus long terminal repeat.

OpenNLP tokenizer. This tokenizer is tested using its plug-in for the GATE framework.

Normal chest x-ray.

2-year 2-month old female with pneumonia.

This may occur through the ability of IL-10 to induce expression of the gene, suppressor of cytokine cueing 3 (SOCS3).

The results identify functionally distinct epitopes on the CD4 coreceptor involved in activation of the **Ras/protein** kinase C and calcium pathways.

The maximal effect is observed at the IL-10 concentration of 20 **U/ml**.

These results indicate that within the **TCR/CD3** cue transduction pathway both PKC and calcineurin are required for the effective activation of the IKK complex and NF-kappaB in T lymphocytes.

Small ^,^ scarred ^ right ^ kidney ^,^ below ^ more ^ than ^ 2 ^ standard ^ deviations ^ in ^ size ^ for ^ patient ^ 's ^ age ^.

If ^ both ^ the ^ best ^ hits ^ of ^ the ^ N- ^ and ^ C-terminal ^ parts ^ are ^ statistically ^ significant ^ (^ E-value ^ **<** ^ 10-5 ^) ^,^ and ^ distance ^ between ^ them ^ is ^ less ^ than ^ 1 ^ kbp ^,^ we ^ call ^ these ^ hits ^ **'syntenic ^ hits ^ '** ^.

The ^ false ^ positive ^ rate ^ (^ FPR ^) ^ of ^ our ^ predictor ^ was ^ estimated ^ by ^ the ^ method ^ of ^ D'Haeseleer ^ and ^ Church ^ 1855 ^ and ^ used ^ to ^ compare ^ it ^ to ^ other ^ prediction ^ datasets ^.

Of ^ these ^,^ Diap1 ^ has ^ been ^ most ^ extensively ^ characterized ^; ^ it ^ can ^ block ^ cell ^ death ^ caused ^ by ^ the ^ ectopic ^ expression ^ of ^ reaper ^,^ hid ^,^ and ^ grim ^ (^ reviewed ^ in ^ **[26]** ^) ^.

These ^ results ^ reveal ^ a ^ central ^ role ^ for ^ **CaMKIV/Gr** ^ as ^ a ^ **Ca ^ (2+)-regulated** ^ activator ^ of ^ gene ^ transcription ^ in ^ T ^ lymphocytes ^.

Two ^ stop ^ codons ^ of ^ an ^ iORF ^ (^ **i ^.e ^.** ^ the ^ inframe ^ and ^ C-terminal ^ stops ^) ^ can ^ be ^ any ^ combination ^ of ^ canonical ^ stop ^ codons ^ (^ TAA ^,^ TAG ^,^ TGA ^) ^.

Selenocysteine ^ and ^ pyrrolysine ^ are ^ the ^ 21st ^ and ^ 22nd ^ amino ^ acids ^,^ which ^ are ^ genetically ^ encoded ^ by ^ stop ^ codons ^.

A ^ total ^ of ^ 26,003 ^ iORF ^ satisfied ^ the ^ above ^ criteria ^.

The ^ patient ^ had ^ prior ^ x-ray ^ on ^ 1/2 ^ which ^ demonstrated ^ no ^ pneumonia ^.

Indeed ^,^ it ^ has ^ been ^ estimated ^ recently ^ that ^ the ^ current ^ yeast ^ and ^ human ^ protein ^ interaction ^ maps ^ are ^ only ^ **50 ^ %** ^ and ^ **10 ^ %** ^ complete ^,^ respectively ^ 18 ^.

The ^ dotted ^ line ^ indicates ^ significance ^ level ^ 0.05 ^ after ^ a ^ correction ^ for ^ multiple ^ testing ^.

E-selectin ^ is ^ induced ^ within ^ 1-2 ^ h ^,^ peaks ^ at ^ 4-6 ^ h ^,^ and ^ gradually ^ returns ^ to ^ basal ^ level ^ by ^ 24 ^ h ^.

2.

1 ^. ^ Bioactivation ^ of ^ sulphamethoxazole ^ (^ SMX ^) ^ to ^ chemically-reactive ^ metabolites ^ and ^ subsequent ^ protein ^ conjugation ^ is ^ thought ^ to ^ be ^ involved ^ in ^ SMX ^ hypersensitivity ^.

Mutants ^ in ^ Toll ^ cueing ^ pathway ^ were ^ obtained ^ from ^ Dr. ^ S. ^ Govind ^: ^ **cactE8,cactIIIG** ^,^ and ^ cactD13 ^ mutations ^ in ^ the ^ cact ^ gene ^ on ^ Chromosome ^ II ^.

The ^ transcripts ^ were ^ detected ^ in ^ all ^ the ^ CD4- ^ CD8- ^,^ CD4+ ^ CD8+ ^,^ CD4+ ^ CD8- ^,^ and ^ CD4- ^ CD8+ ^ cell ^ populations ^.

Footprinting ^ analysis ^ revealed ^ that ^ the ^ identical ^ sequence ^ CCGAAACTGAAAAGG ^,^ designated ^ E6 ^,^ was ^ protected ^ by ^ nuclear ^ extracts ^ from ^ B ^ cells ^,^ T ^ cells ^,^ or ^ HeLa ^ cells ^.

Bcd ^ mRNA ^ transcripts ^ of ^ **<** ^ or ^ = ^ 2.6 ^ kb ^ were ^ selectively ^ expressed ^ in ^ PBL ^ and ^ testis ^ of ^ healthy ^ individuals ^.

Names ^ of ^ all ^ available ^ Trace ^ Databases ^ were ^ taken ^ from ^ a ^ list ^ of ^ databases ^ at ^ **http ^://www.ncbi.nlm.nih.gov/blast/mmtrace.shtml**

This was last documented on the Nuclearv Cystogram dated 1/2/01.

We found IL-2Ralpha expression to be increased in BAL cells from involved sites of active pulmonary tuberculosis.

Expression of a highly specific protein inhibitor for cyclic AMP-dependent protein kinases in interleukin-1 **(IL-1)-responsive** cells blocked IL-1-induced gene transcription that was driven by the kappa immunoglobulin enhancer or the human immunodeficiency virus long terminal repeat.

MedPost tokenizer. It has been tested using its java version through command line.

Normal chest **x-ray**.

2-year 2-month old female with pneumonia.

This may occur through the ability of **IL-10** to induce expression of the gene, suppressor of cytokine cueing 3 (SOCS3).

The results identify functionally distinct epitopes on the CD4 coreceptor involved in activation of the Ras/protein kinase C and calcium pathways.

The maximal effect is observed at the **IL-10** concentration of 20 U/ml.

These results indicate that within the TCR/CD3 cue transduction pathway both PKC and calcineurin are required for the effective activation of the IKK complex and **NF-kappaB** in T lymphocytes.

Small, scarred right kidney, below more than 2 standard deviations in size for **patient's** age.

If both the best hits of the **N-** and **C-terminal** parts are statistically significant (**E-value <10-5**), and distance between them is less than 1 kbp, we call these hits 'syntenic hits'.

The false positive rate (FPR) of our predictor was estimated by the method of **D'Haeseleer** and Church 1855 and used to compare it to other prediction datasets.

Of these, Diap1 has been most extensively characterized; it can block cell death caused by the ectopic expression of reaper, hid, and grim (reviewed in [26]).

These results reveal a central role for CaMKIV/Gr as a **Ca(2+)-regulated** activator of gene transcription in T lymphocytes.

Two stop codons of an iORF (**i.e.**, the inframe and **C-terminal** stops) can be any combination of canonical stop codons (TAA, TAG, TGA).

Selenocysteine and pyrrolysine are the 21st and 22nd amino acids, which are genetically encoded by stop codons.

A total of **26,003** iORF satisfied the above criteria.

The^patient^had^prior^**x^-^ray**^on^**1^/^2**^which^demonstrated^no^pneumonia^.

Indeed^,^it^has^been^estimated^recently^that^the^current^yeast^and^human^protein^interaction^maps^are^only^**50^%**^and^**10^%**^complete^,^respectively^18^.

The^dotted^line^indicates^significance^level^**0^.^05**^after^a^correction^for^multiple^testing^.

E-selectin^is^induced^within^**1^-^2**^h^,^peaks^at^**4^-^6**^h^,^and^gradually^returns^to^basal^level^by^24^h^.

2^.

1^.^Bioactivation^of^sulphamethoxazole^(^SMX^)^to^**chemically^-^reactive**^metabolites^and^subsequent^protein^conjugation^is^thought^to^be^involved^in^SMX^hypersensitivity^.

Mutants^in^Toll^cueing^pathway^were^obtained^from^**Dr^.^S^.^**Govind^:^cactE8^,^cactIIIG^,^and^cactD13^mutations^in^the^cact^gene^on^Chromosome^II^.

The^transcripts^were^detected^in^all^the^**CD4^-^CD8^-^,^CD4^+^CD8^+^,^CD4^+^CD8^-^,^and^CD4^-^CD8^+**^cell^populations^.

Footprinting^analysis^revealed^that^the^identical^sequence^CCGAAACTGAAAAGG^,^designated^E6^,^was^protected^by^nuclear^extracts^from^B^cells^,^T^cells^,^or^HeLa^cells^.

Bcd^mRNA^transcripts^of^**&^lt^;^**or^=^**2^.^6**^kb^were^selectively^expressed^in^PBL^and^testis^of^healthy^individuals^.

Names^of^all^available^Trace^Databases^were^taken^from^a^list^of^databases^at^**http^:^/^/^www^.^ncbi^.^nlm^.^nih^.^gov^/^blast^/^mmtrace^.^shtml**

This^was^last^documented^on^the^Nuclearv^Cystogram^dated^**1^/2^/^01**^.

We^found^**IL^-^2Ralpha**^expression^to^be^increased^in^BAL^cells^from^involved^sites^of^active^pulmonary^tuberculosis^.

Expression^of^a^highly^specific^protein^inhibitor^for^cyclic^**AMP-^dependent**^protein^kinases^in^**interleukin^-1^(^IL-1^)^-^responsive**^cells^blocked^**IL^-^1^-^induced**^gene^transcription^that^was^driven^by^the^kappa^immunoglobulin^enhancer^or^the^human^immunodeficiency^virus^long^terminal^repeat^.

Stanford POS tagger. It is tested using the Java API.

Normal^chest^x-ray^.

2-year^2-month^old^female^with^pneumonia^.

This^may^occur^through^the^ability^of^IL-10^to^induce^expression^of^the^gene^,^suppressor^of^cytokine^cueing^3^(^SOCS3^)^.

The^results^identify^functionally^distinct^epitopes^on^the^CD4^coreceptor^involved^in^activation^of^the^**Ras/protein**^kinase^C^and^calcium^pathways^.

The^maximal^effect^is^observed^at^the^IL-10^concentration^of^20^**U/ml**^.

These results indicate that within the **TCR/CD3** cue transduction pathway both PKC and calcineurin are required for the effective activation of the IKK complex and NF-kappaB in T lymphocytes.

Small, scarred right kidney, below more than 2 standard deviations in size for patient 's age.

If both the best hits of the N- and C-terminal parts are statistically significant (E-value < 10-5), and distance between them is less than 1 kbp, we call these hits 'syntenic hits'.

The false positive rate (FPR) of our predictor was estimated by the method of D'Haeseleer and Church 1855 and used to compare it to other prediction datasets.

Of these, Diap1 has been most extensively characterized; it can block cell death caused by the ectopic expression of reaper, hid, and grim (reviewed in [26]).

These results reveal a central role for **CaMKIV/Gr** as a **Ca (2+) -regulated** activator of gene transcription in T lymphocytes.

Two stop codons of an iORF (i.e. the inframe and C-terminal stops) can be any combination of canonical stop codons (TAA, TAG, TGA).

Selenocysteine and pyrrolysine are the 21st and 22nd amino acids, which are genetically encoded by stop codons.

A total of 26,003 iORF satisfied the above criteria.

The patient had prior x-ray on 1/2 which demonstrated no pneumonia.

Indeed, it has been estimated recently that the current yeast and human protein interaction maps are only **50%** and **10%** complete, respectively 18.

The dotted line indicates significance level 0.05 after a correction for multiple testing.

E-selectin is induced within 1-2 h, peaks at 4-6 h, and gradually returns to basal level by 24 h.

2.

1. Bioactivation of sulphamethoxazole (SMX) to chemically-reactive metabolites and subsequent protein conjugation is thought to be involved in SMX hypersensitivity.

Mutants in Toll cueing pathway were obtained from Dr. S. Govind: cactE8, cactIIIG, and cactD13 mutations in the cact gene on Chromosome II.

The transcripts were detected in all the **CD4-CD8-**, **CD4+CD8+**, **CD4+CD8-**, and **CD4-CD8+** cell populations.

Footprinting analysis revealed that the identical sequence CCGAAACTGAAAAGG, designated E6, was protected by nuclear extracts from B cells, T cells, or HeLa cells.

Bcd mRNA transcripts of < or = 2.6 kb were selectively expressed in PBL and testis of healthy individuals.

Names of all available Trace Databases were taken from a list of databases at http://www.ncbi.nlm.nih.gov/blast/mmtrace.shtml

This was last documented on the Nuclearv Cystogram dated 1/2/01.

We found IL-2Ralpha expression to be increased in BAL cells from involved sites of active pulmonary tuberculosis.

Expression of a highly specific protein inhibitor for cyclic AMP-dependent protein kinases in interleukin-1**(IL-1)-responsive** cells blocked IL-1-induced gene transcription that was driven by the kappa immunoglobulin enhancer or the human immunodeficiency virus long terminal repeat.

Freeling. The online demo has been used to test it.

Normal chest x-ray.

2-year 2-month old female with pneumonia.

This may occur through the ability of IL-10 to induce expression of the gene, suppressor of cytokine cueing 3 (SOCS3).

The results identify functionally distinct epitopes on the CD4 coreceptor involved in activation of the Ras/protein kinase C and calcium pathways.

The maximal effect is observed at the IL-10 concentration of 20 U/**ml.**

These results indicate that within the **TCR/CD3** cue transduction pathway both PKC and calcineurin are required for the effective activation of the IKK complex and NF-kappaB in T lymphocytes.

Small, scarred right kidney, below more than 2 standard deviations in size for patient's age.

If both the best hits of the **N-** and C-terminal parts are statistically significant (E-value 10-5), and distance between them is less than 1 kbp, we call these hits 'syntenic hits'.

The false positive rate (FPR) of our predictor was estimated by the method of D'Haeseleer and Church 1855 and used to compare it to other prediction datasets.

Of these, Diap1 has been most extensively characterized; it can block cell death caused by the ectopic expression of reaper, hid, and grim (reviewed in [26]).

These results reveal a central role for CaMKIV/Gr as a **Ca(2+)-regulated** activator of gene transcription in T lymphocytes.

Two stop codons of an iORF (i.e. the inframe and C-terminal stops) can be any combination of canonical stop codons (TAA, TAG, TGA).

Selenocysteine and pyrrolysine are the 21st and 22nd amino acids, which are genetically encoded by stop codons.

A total of 26,003 iORF satisfied the above criteria.

The patient had prior x-ray on 1/2 which demonstrated no pneumonia.

Indeed, it has been estimated recently that the current yeast and human protein interaction maps are only **50_%** and **10_%** complete, respectively 18.

The dotted line indicates significance level 0.05 after a correction for multiple testing.

E-selectin is induced within 1-2 h, peaks at 4-6 h, and gradually returns to basal level by 24 h.

2.

1. Bioactivation of sulphamethoxazole (SMX) to chemically-reactive metabolites and subsequent protein conjugation is thought to be involved in SMX hypersensitivity.

Mutants in Toll cueing pathway were obtained from **Dr._S._Govind**: **cactE8,cactIIIG**, and cactD13 mutations in the cact gene on **Chromosome_II**.

The transcripts were detected in all the **CD4-CD8-**, **CD4+CD8+**, **CD4+CD8-**, and **CD4-CD8+** cell populations.

Footprinting analysis revealed that the identical sequence CCGAAACTGAAAAGG, designated E6, was protected by nuclear extracts from B cells, T cells, or HeLa cells.

Bcd mRNA transcripts of or = 2.6 kb were selectively expressed in PBL and testis of healthy individuals.

Names of all available **Trace_Databases** were taken from a list of databases at **http://www.ncbi.nlm.nih.gov/blast/mmtrace.shtml**

This was last documented on the **Nuclearv_Cystogram** dated 1/2/01.

We found IL-2Ralpha expression to be increased in BAL cells from involved sites of active pulmonary tuberculosis.

Expression of a highly specific protein inhibitor for cyclic AMP-dependent protein kinases in interleukin-1 **(IL-1)-responsive** cells blocked IL-1-induced gene transcription that was driven by the kappa immunoglobulin enhancer or the human immunodeficiency virus long terminal repeat.

LingPipe. This tool is tested through its plug-in for GATE.

Normal chest **x-ray**.

2-year 2-month old female with pneumonia.

This may occur through the ability of **IL-10** to induce expression of the gene, suppressor of cytokine cueing 3 (SOCS3).

The results identify functionally distinct epitopes on the CD4 coreceptor involved in

activation of the Ras/protein kinase C and calcium pathways.

The maximal effect is observed at the **IL-10** concentration of 20 U/ml.

These results indicate that within the TCR/CD3 cue transduction pathway both PKC and calcineurin are required for the effective activation of the IKK complex and **NF-kappaB** in T lymphocytes.

Small, scarred right kidney, below more than 2 standard deviations in size for **patient's** age.

If both the best hits of the **N-** and **C-terminal** parts are statistically significant (**E-value <10-5**), and distance between them is less than 1 kbp, we call these 'syntenic hits'.

The false positive rate (FPR) of our predictor was estimated by the method of **D' Haeseleer** and Church 1855 and used to compare it to other prediction datasets.

Of these, Diap1 has been most extensively characterized; it can block cell death caused by the ectopic expression of reaper, hid, and grim (reviewed in [26]).

These results reveal a central role for CaMKIV/Gr as a **Ca(2+)-regulated** activator of gene transcription in T lymphocytes.

Two stop codons of an iORF (**i.e.** the inframe and C-terminal stops) can be any combination of canonical stop codons (TAA, TAG, TGA).

Selenocysteine and pyrrolysine are the 21st and 22nd amino acids, which are genetically encoded by stop codons.

A total of 26,003 iORF satisfied the above criteria.

The patient had prior **x-ray** on **1/2** which demonstrated no pneumonia.

Indeed, it has been estimated recently that the current yeast and human protein interaction maps are only **50%** and **10%** complete, respectively 18.

The dotted line indicates significance level 0.05 after a correction for multiple testing.

E-selectin is induced within **1-2** h, peaks at **4-6** h, and gradually returns to basal level by 24 h.

2.

1. Bioactivation of sulphamethoxazole (SMX) to **chemically-reactive** metabolites and subsequent protein conjugation is thought to be involved in SMX hypersensitivity.

Mutants in Toll cueing pathway were obtained from **Dr. S.** Govind: cactE8, cactIIIG, and cactD13 mutations in the cact gene on Chromosome II.

The transcripts were detected in all the CD4-**CD8-**, **CD4+CD8+**, **CD4+CD8-**, and **CD4-CD8+** cell populations.

Footprinting analysis revealed that the identical sequence CCGAAACTGAAAAGG, designated

E6 , was protected by nuclear extracts from B cells , T cells , or HeLa cells .

Bcd mRNA transcripts of **<** or = 2.6 kb were selectively expressed in PBL and testis of healthy individuals .

Names of all available Trace Databases were taken from a list of databases at **http://www.ncbi.nlm.nih.gov/blast/mmtrace.shtml**

This was last documented on the Nuclearv Cystogram dated **1/2/01** .

We found **IL-2Ralpha** expression to be increased in BAL cells from involved sites of active pulmonary tuberculosis .

Expression of a highly specific protein inhibitor for cyclic AMP-dependent protein kinases in **interleukin-1 (IL-1) -responsive** cells blocked **IL-1-induced** gene transcription that was driven by the kappa immunoglobulin enhancer or the human immunodeficiency virus long terminal repeat .

JULIE LAB tokenizer. It has been tested using its java version through command line.

Normal chest **x-ray** .

2-year 2-month old female with pneumonia .

This may occur through the ability of IL-10 to induce expression of the gene , suppressor of cytokine cueing 3 (SOCS3) .

The results identify functionally distinct epitopes on the CD4 coreceptor involved in activation of the Ras / protein kinase C and calcium pathways .

The maximal effect is observed at the IL-10 concentration of 20 U / ml .

These results indicate that within the TCR / CD3 cue transduction pathway both PKC and calcineurin are required for the effective activation of the IKK complex and NF-kappaB in T lymphocytes .

Small , scarred right kidney , below more than 2 standard deviations in size for patient 's age .

If both the best hits of the **N-** and **C-terminal** parts are statistically significant (E-value **<** 10-5) , and distance between them is less than 1 kbp , we call these hits ' syntenic hits ' .

The false positive rate (FPR) of our predictor was estimated by the method of D'Haeseleer and Church 1855 and used to compare it to other prediction datasets .

Of these , Diap1 has been most extensively characterized ; it can block cell death caused by the ectopic expression of reaper , hid , and grim (reviewed in **[26])** .

These results reveal a central role for CaMKIV / Gr as a **Ca(2+)-regulated** activator of gene transcription in T lymphocytes .

Two stop codons of an iORF (i.e. the inframe and **C-terminal** stops) can be any combination of canonical stop codons (TAA , TAG , TGA) .

Selenocysteine and pyrrolysine are the 21st and 22nd amino acids, which are genetically encoded by stop codons.

A total of **26,003** iORF satisfied the above criteria.

The patient had prior **x-ray** on **1/2** which demonstrated no pneumonia.

Indeed, it has been estimated recently that the current yeast and human protein interaction maps are only **50%** and **10%** complete, respectively 18.

The dotted line indicates significance level 0.05 after a correction for multiple testing.

E-selectin is induced within **1-2** h, peaks at **4-6** h, and gradually returns to basal level by 24 h.

2.

1. Bioactivation of sulphamethoxazole (SMX) to **chemically-reactive** metabolites and subsequent protein conjugation is thought to be involved in SMX hypersensitivity.

Mutants in Toll cueing pathway were obtained from Dr. S. Govind: cactE8, cactIIIG, and cactD13 mutations in the cact gene on Chromosome II.

The transcripts were detected in all the **CD4-CD8-**, **CD4+CD8+**, **CD4+CD8-**, and **CD4-CD8+** cell populations.

Footprinting analysis revealed that the identical sequence CCGAAACTGAAAAGG, designated E6, was protected by nuclear extracts from B cells, T cells, or HeLa cells.

Bcd mRNA transcripts of **<** or = 2.6 kb were selectively expressed in PBL and testis of healthy individuals.

Names of all available Trace Databases were taken from a list of databases at **http://www.ncbi.nlm.nih.gov/blast/mmtrace.shtml l**

This was last documented on the Nuclearv Cystogram dated **1/2/01**.

We found IL-2Ralpha expression to be increased in BAL cells from involved sites of active pulmonary tuberculosis.

Expression of a highly specific protein inhibitor for cyclic **AMP-dependent** protein kinases in **interleukin-1 (IL-1)-responsive** cells blocked **IL-1-induced** gene transcription that was driven by the kappa immunoglobulin enhancer or the human immunodeficiency virus long terminal repeat.

Penn Bio tokenizer. This tokenizer is tested using the plug-in for GATE.

Normal chest x-ray.

2-year 2-month old female with pneumonia.

This may occur through the ability of IL-10 to induce expression of the gene, suppressor of cytokine cueing 3 (SOCS3).

The results identify functionally distinct epitopes on the CD4 coreceptor involved in activation of the Ras/protein kinase C and calcium pathways.

The maximal effect is observed at the IL-10 concentration of 20 U/ml.

These results indicate that within the TCR/CD3 cue transduction pathway both PKC and calcineurin are required for the effective activation of the IKK complex and NF-kappaB in T lymphocytes.

Small, scarred right kidney, below more than 2 standard deviations in size for patient's age.

If both the best hits of the **N-** and **C-terminal** parts are statistically significant (**E-value <10-5**), and distance between them is less than 1 kbp, we call these hits 'syntenic hits'.

The false positive rate (FPR) of our predictor was estimated by the method of D'Haeseleer and Church 1855 and used to compare it to other prediction datasets.

Of these, Diap1 has been most extensively **characterized**; it can block cell death caused by the ectopic expression of reaper, hid, and grim (reviewed in [26]).

These results reveal a central role for CaMKIV/Gr as a **Ca(2+)-regulated** activator of gene transcription in T lymphocytes.

Two stop codons of an iORF (i.e. the inframe and **C-terminal** stops) can be any **combination** of canonical stop codons (TAA, TAG, TGA).

Selenocysteine and pyrrolysine are the **21st** and 22nd amino acids, which are genetically **encoded** by stop codons.

A total of 26,003 iORF satisfied the above criteria.

The patient had prior x-ray on **1/2** which demonstrated no pneumonia.

Indeed, it has been estimated recently that the current yeast and human protein interaction maps are only **50%** and **10%** complete, respectively 18.

The dotted line indicates significance level 0.05 after a correction for multiple testing.

E-selectin is induced within 1-2 h, peaks at 4-6 h, and gradually returns to basal level by 24 h.

2.

1. Bioactivation of sulphamethoxazole (SMX) to **chemically-reactive** metabolites and subsequent protein conjugation is thought to be involved in SMX hypersensitivity.

Mutants in Toll cueing pathway were **obtained** from **Dr**. S. Govind: cactE8, **cactIIIG**, and cactD13 mutations in the cact gene on Chromosome II.

The transcripts were detected in all the **CD4-CD8-**, **CD4+CD8+**, **CD4+CD8-**, and **CD4-CD8+** cell populations.

Footprinting^analysis^revealed^that^the^identical^sequence^CCGAAACTGAAAAGG^,^**design^ated** ^E6^,^was^protected^by^nuclear^extracts^from^B^cells^,^T^cells^,^or^HeLa^cells^.

Bc^d^mRNA^transcripts^of^**&lt^;^**or^=^2.6^kb^were^selectively^expressed^in^PBL^and^testis ^of^healthy^individuals^.

Names^of^all^available^Trace^Databases^were^taken^from^a^list^of^databases^at^ **http^:^/^/^www.ncbi.nlm.nih.gov/^blast^/^mmtrace^.^shtml**

This^was^last^documented^on^the^Nuclearv^Cystogram^dated^**1^/^2^/^01**^.

We^found^IL-2Ralpha^expression^to^be^increased^in^BAL^cells^from^involved^sites^of^ active^pulmonary^tuberculosis^.

Expression^of^a^highly^specific^protein^inhibitor^for^cyclic^**AMP^-^dependent**^protein^ kinases^in^interleukin-1^**(^IL-1)^-^responsive**^cells^blocked^**IL-1^-^induced**^gene^ transcription^that^was^driven^by^the^**ka^ppa**^immunoglobulin^enhancer^or^the^human^ immunodeficiency^virus^long^terminal^repeat^.

Gate Unicode tokenizer. It has been tested using the plug-in for GATE.

Normal^chest^x-ray^.

2^-^year^2^-^month^old^female^with^pneumonia^.

This^may^occur^through^the^ability^of^**IL-^10**^to^induce^expression^of^the^gene^,^ suppressor^of^cytokine^cueing^3^(^SOCS3^)^.

The^results^identify^functionally^distinct^epitopes^on^the^**CD^4**^coreceptor^involved^in^ activation^of^the^Ras^/^protein^kinase^C^and^calcium^pathways^.

The^maximal^effect^is^observed^at^the^**IL-^10**^concentration^of^20^U^/^ml^.

These^results^indicate^that^within^the^TCR^/^CD^3^cue^transduction^pathway^both^PKC^ and^calcineurin^are^required^for^the^effective^activation^of^the^IKK^complex^and^NF-kappaB^in^T^lymphocytes^.

Small^,^scarred^right^kidney^,^below^more^than^2^standard^deviations^in^size^for^**patient^' ^s**^age^.

If^both^the^best^hits^of^the^N-^and^**C-^terminal**^parts^are^statistically^significant^(^**E-^ value^&^lt^;^10^-^5**^)^,^and^distance^between^them^is^less^than^1^kbp^,^we^call^these^ hits^'^syntenic^hits^'^.

The^false^positive^rate^(^FPR^)^of^our^predictor^was^estimated^by^the^method^of^**D^' ^ Haeseleer**^and^Church^1855^and^used^to^compare^it^to^other^prediction^ datasets^.

Of^these^,^**Diap^1**^has^been^most^extensively^characterized^;^it^can^block^cell^death^ caused^by^the^ectopic^expression^of^reaper^,^hid^,^and^grim^(^reviewed^ in^[^26^]^)^.

These^results^reveal^a^central^role^for^CaMKIV^/^Gr^as^a^**Ca^(^2+^)^-^regulated**^ activator^of^gene^ transcription^in^T^lymphocytes^.

Two stop codons of an iORF (**i.e.** the inframe and **C-terminal** stops) can be any combination of canonical stop codons (TAA, TAG, TGA).

Selenocysteine and pyrrolysine are the **21st** and **22nd** amino acids, which are genetically encoded by stop codons.

A total of **26,003** iORF satisfied the above criteria.

The patient had prior x-ray on **1/2** which demonstrated no pneumonia.

Indeed, it has been estimated recently that the current yeast and human protein interaction maps are only **50%** and **10%** complete, respectively 18.

The dotted line indicates significance level **0.05** after a correction for multiple testing.

E-selectin is induced within **1-2** h, peaks at **4-6** h, and gradually returns to basal level by 24 h.

2.

1. Bioactivation of sulphamethoxazole (SMX) to chemically-reactive metabolites and subsequent protein conjugation is thought to be involved in SMX hypersensitivity.

Mutants in Toll cueing pathway were obtained from **Dr.S.** Govind: **cactE8**, cactIIIG, and **cactD13** mutations in the cact gene on Chromosome II.

The transcripts were detected in all the **CD4-CD8-**, **CD4+CD8+**, **CD4+CD8-**, and **CD4-CD8+** cell populations.

Footprinting analysis revealed that the identical sequence CCGAAACTGAAAAGG, designated **E6**, was protected by nuclear extracts from B cells, T cells, or HeLa cells.

Bcd mRNA transcripts of **<** or = **2.6** kb were selectively expressed in PBL and testis of healthy individuals.

Names of all available Trace Databases were taken from a list of databases at **http://www.ncbi.nlm.nih.gov/blast/mmtrace.shtml**

This was last documented on the Nuclearv Cystogram dated **1/2/01**.

We found **IL-2Ralpha** expression to be increased in BAL cells from involved sites of active pulmonary tuberculosis.

Expression of a highly specific protein inhibitor for cyclic AMP-dependent protein kinases in **interleukin-1(IL-1)-responsive** cells blocked **IL-1-induced** gene transcription that was driven by the kappa immunoglobulin enhancer or the human immunodeficiency virus long terminal repeat.

Genia Tagger. The online demo has been used to test it.

Normal chest x-ray.

2-year 2-month old female with pneumonia.

This^may^occur^through^the^ability^of^IL-10^to^induce^expression^of^the^gene^,^ suppressor^of^cytokine^cueing^3^(^SOCS3^)^.

The^results^identify^functionally^distinct^epitopes^on^the^CD4^coreceptor^involved^in^ activation^of^the^**Ras/protein**^kinase^C^and^calcium^pathways^.

The^maximal^effect^is^observed^at^the^IL-10^concentration^of^20^**U/ml**^.

These^results^indicate^that^within^the^**TCR/CD3**^cue^transduction^pathway^both^PKC^and^ca lcineurin^are^required^for^the^effective^activation^of^the^IKK^complex^and^NF-kappaB^in^T^lymphocytes^.

Small^,^scarred^right^kidney^,^below^more^than^2^standard^deviations^in^size^for^patient^'s ^age^.

If^both^the^best^hits^of^the^N-^and^C-terminal^parts^are^statistically^significant^(^E-value^ **&^lt^;**^10-5^)^,^and^distance^between^them^is^less^than^1^kbp^,^we^call^these^hits^ **'syntenic^hits^'**^.

The^false^positive^rate^(^FPR^)^of^our^predictor^was^estimated^by^the^method^of^ D'Haeseleer^and^Church^1855^and^used^to^compare^it^to^other^prediction^datasets^.

Of^these^,^Diap1^has^been^most^extensively^characterized^;^it^can^block^cell^death^caused^ by^the^ectopic^expression^of^reaper^,^hid^,^and^grim^(^reviewed^in^[^26^]^)^.

These^results^reveal^a^central^role^for^**CaMKIV/Gr**^as^a^**Ca^(^2+^)^-regulated**^activator^ of^gene^ transcription^in^T^lymphocytes^.

Two^stop^codons^of^an^iORF^(^**i.e^.**^the^inframe^and^C-terminal^stops^)^can^be^any^ combination^of^ canonical^stop^codons^(^TAA^,^TAG^,^TGA^)^.

Selenocysteine^and^pyrrolysine^are^the^21st^and^22nd^amino^acids^,^which^are^ genetically^ encoded^by^stop^codons^.

A^total^of^**26^,^003**^iORF^satisfied^the^above^criteria^.

The^patient^had^prior^x-ray^on^1/2^which^demonstrated^no^pneumonia^.

Indeed^,^it^has^been^estimated^recently^that^the^current^yeast^and^human^protein^ interaction^maps^are^only^**50^%**^and^**10^%**^complete^,^respectively^18^.

The^dotted^line^indicates^significance^level^0.05^after^a^correction^for^multiple^ testing^.

E-selectin^is^induced^within^1-2^h^,^peaks^at^4-6^h^,^and^gradually^returns^to^basal^ level^by^24^h^.

2^.

1^.^Bioactivation^of^sulphamethoxazole^(^SMX^)^to^chemically-reactive^metabolites^and^ subsequent^protein^conjugation^is^thought^to^be^involved^in^SMX^hypersensitivity^.

Mutants^in^Toll^cueing^pathway^were^obtained^from^**Dr^.^S^.**^Govind^:^cactE8^,^cactIIIG^,^ and^cactD13^mutations^in^the^cact^gene^on^Chromosome^II^.

The transcripts were detected in all the CD4- CD8- , CD4+ CD8+ , CD4+ CD8- , and CD4- CD8+ cell populations.

Footprinting analysis revealed that the identical sequence CCGAAACTGAAAAGG , designated E6 , was protected by nuclear extracts from B cells , T cells , or HeLa cells.

Bcd mRNA transcripts of **<** or =2.6 kb were selectively expressed in PBL and testis of healthy individuals.

Names of all available Trace Databases were taken from a list of databases at **http://www.ncbi.nlm.nih.gov/blast/mmtrace.shtml**

This was last documented on the Nuclearv Cystogram dated 1/2/01.

We found IL-2Ralpha expression to be increased in BAL cells from involved sites of active pulmonary tuberculosis.

Expression of a highly specific protein inhibitor for cyclic AMP-dependent protein kinases in interleukin-1 **(IL-1) -responsive** cells blocked IL-1-induced gene transcription that was driven by the kappa immunoglobulin enhancer or the human immunodeficiency virus long terminal repeat.

ERG – Online Resource Grammar. This tool is tested using the online demo.

Normal chest **x-ray**.

-

This may occur through the ability of IL-10 to induce expression of the **gene,** suppressor of cytokine cueing 3 **(SOCS3)**.

The results identify functionally distinct epitopes on the CD4 coreceptor involved in activation of the Ras / protein kinase C and calcium **pathways.**

The maximal effect is observed at the IL-10 concentration of 20 U / ml.

-

-

-

The false positive rate **(FPR)** of our predictor was estimated by the method of D'Haeseleer and Church 1855 and used to compare it to other prediction **datasets.**

-

These results reveal a central role for CaMKIV / Gr as a **Ca(2+)-regulated** activator of gene transcription in T lymphocytes.

Two stop codons of an iORF **(i.e.** the inframe and **C-terminal stops)** can be any combination of canonical stop codons (**TAA, TAG, TGA).**

Selenocysteine^and^pyrrolysine^are^the^21st^and^22nd^amino^**acids,**^which^are^genetically ^encoded^by^stop^**codons.**

A^total^of^26,003^iORF^satisfied^the^above^**criteria.**

The^patient^had^prior^**x-^ray**^on^1/2^which^demonstrated^no^**pneumonia.**

-

The^dotted^line^indicates^significance^level^0.05^after^a^correction^for^multiple^ **testing.**

-

2.

-

-

-

Footprinting^analysis^revealed^that^the^identical^sequence^**CCGAAACTGAAAAGG,**^designated^ **E6,**^was^protected^by^nuclear^extracts^from^B^**cells,**^T^**cells,**^or^HeLa^**cells.**

-

Names^of^all^available^Trace^Databases^were^taken^from^a^list^of^databases^at^ http://www.ncbi.nlm.nih.gov/blast/mmtrace.shtml

This^was^last^documented^on^the^Nuclearv^Cystogram^dated^1/2/01^.

-

-

Xerox tokenizer. Its demo online has been used to test this tokenizer.

Normal^chest^**x-^ray**.

2-year^2-month^old^female^with^pneumonia^.

This^may^occur^through^the^ability^of^IL-10^to^induce^expression^of^the^gene^,^suppressor ^of^cytokine^cueing^3^(^ SOCS3^)^.

The^results^identify^functionally^distinct^epitopes^on^the^CD4^coreceptor^involved^in^ activation^of^the^**Ras/protein**^kinase^C^and^calcium^pathways^.

The^maximal^effect^is^observed^at^the^IL-10^concentration^of^20^**U/ml**^.

These^results^indicate^that^within^the^**TCR/CD3**^cue^transduction^pathway^both^PKC^and^ca lcineurin^are^required^for^the^effective^activation^of^the^IKK^complex^and^NF-kappaB^ in^T^lymphocytes^.

Small , scarred right kidney , below more than 2 standard deviations in size for patient 's age .

If both the best hits of the N- and C-terminal parts are statistically significant (E-value **<** 10-5) , and distance between them is less than 1 kbp , we call these hits ' syntenic hits ' .

The false positive rate (FPR) of our predictor was estimated by the method of D'Haeseleer and Church 1855 and used to compare it to other prediction datasets .

Of these , Diap1 has been most extensively characterized ; it can block cell death caused by the ectopic expression of reaper , hid , and grim (reviewed in [26]) .

These results reveal a central role for **CaMKIV/Gr** as a **Ca (2+)-regulated** activator of gene transcription in T lymphocytes .

Two stop codons of an iORF (i.e. the inframe and C-terminal stops) can be any combination of canonical stop codons (TAA , TAG , TGA) .

Selenocysteine and pyrrolysine are the 21st and 22nd amino acids , which are genetically encoded by stop codons .

A total of 26,003 iORF satisfied the above criteria .

The patient had prior x-ray on 1/2 which demonstrated no pneumonia .

Indeed , it has been estimated recently that the current yeast and human protein interaction maps are only 50% and 10% complete , respectively 18 .

The dotted line indicates significance level 0.05 after a correction for multiple testing .

E-selectin is induced within 1-2 h , peaks at 4-6 h , and gradually returns to basal level by 24 h .

2 .

1 . Bioactivation of sulphamethoxazole (SMX) to chemically-reactive metabolites and subsequent protein conjugation is thought to be involved in SMX hypersensitivity .

Mutants in Toll cueing pathway were obtained from Dr. S. Govind : **cactE8,cactIIIG** , and cactD13 mutations in the cact gene on Chromosome **II.**

The transcripts were detected in all the CD4- CD8- , CD4+ CD8+ , CD4+ CD8- , and CD4- CD8+ cell populations .

Footprinting analysis revealed that the identical sequence CCGAAACTGAAAAGG , designated E6 , was protected by nuclear extracts from B cells , T cells , or HeLa cells .

Bcd mRNA transcripts of **<** or = 2.6 kb were selectively expressed in PBL and testis of healthy individuals .

Names of all available Trace Databases were taken from a list of databases at http://www.ncbi.nlm.nih.gov/blast/mmtrace.shtml

This^was^last^documented^on^the^Nuclearv^Cystogram^dated^1/2/01^.

We^found^IL-2Ralpha^expression^to^be^increased^in^BAL^cells^from^involved^sites^of^ active^ pulmonary^tuberculosis^.

Expression^of^a^highly^specific^protein^inhibitor^for^cyclic^AMP-dependent^protein^kinases^ in^interleukin-1^**(^IL-1^)-responsive**^cells^blocked^IL-1-induced^gene^transcription^that^was^ driven^by^the^kappa^immunoglobulin^enhancer^or^the^human^immunodeficiency^virus^long^ terminal^repeat^.

Appendix D

Available software used

Weka

Weka[8] (Witten & Frank, 2005) is a popular machine-learning software suite that supports several standard data-mining algorithms. Among its characteristics highlight the following:

- It is freely available under the GNU General Public License.
- It is portable.
- It includes a comprehensive collection of data pre-processing and modelling techniques.
- It is easy to use due to its graphical user interfaces.

Weka supports several standard data mining tasks, more specifically, data pre-processing, clustering, classification, regression, visualization, and feature selection. All of Weka's techniques are predicated on the assumption that the data is available as a single flat file or relation, where each data point is described by a fixed number of attributes (normally, numeric or nominal attributes, but some other attribute types are also supported).

Weka's main user interface is the Explorer, but essentially the same functionality can be accessed through the component-based Knowledge Flow interface and from the command line. There is also the Experimenter, which allows the systematic comparison of the predictive performance of Weka's machine learning algorithms on a collection of datasets.

---

[8] See *http://www.cs.waikato.ac.nz/ml/weka/*

In this thesis, Naïve Bayes and C4.5 algorithms implemented the version 3.6 of Weka have been used in the developed of the negation and speculation detection systems (See chapters 4.2.1 and 5.2.1). In addition, Weka has also been employed to obtain the final feature set which represent all the tokens that appear in the collections of documents for both the biomedical (Chapter 4.2.2) and the review domain (Chapter 5.2.2). To do this, the information gained and chi-squared feature selection techniques implemented in this software have been applied to the initial set of attributes.

LibSVM

LIBSVM[9] (Chang & Lin, 2011) is one of the most widely used library for Support Vector Machines (SVMs) developed at the National Taiwan University. It implements the Sequential Minimal Optimisation (SMO) algorithm (Platt,), i.e., an algorithm for solving the quadratic programming problem that arises during the training of SVM, for kernelized SVMs, supporting classification and regression.

Main features of LIBSVM include:

- Different SVM formulations.
- Efficient multi-class classification.
- Cross validation for model selection.
- Probability estimates.
- Various kernels.
- Weighted SVM for imbalanced data.
- Both C++ and Java sources.
- Interfaces in different programming languages.
- It is also included in some data mining environments such as RapidMiner[10] (Mierswa, Wurst, Klinkenberg, Scholz, & Euler, 2006).

In the development of negation and speculation detection systems presented in this thesis, it has been experimented with SVM implemented in the version 3.16 of this software. Linear, Polynomial, Radial Basis Function (RBF), and Sigmoid kernels have been tested in the

[9] See *http://www.csie.ntu.edu.tw/~cjlin/libsvm/*
[10] See *http://rapidminer.com/*

biomedical domain as detailed in the Chapter 4.2.1 and RBF kernel in the review domain (See Chapter 5.2.1).

Stanford POS tagger

The Stanford POS tagger[11] (Toutanova, Klein et al. 2003) is a Java implementation of the log-linear POS taggers originally written by Kristina Toutanova. It is developed at the Stanford University.

The POS returned by this parser has been used in this thesis as one of the features employed in the negation and speculation detection approaches for both the biomedical (Chapter 4.2.2) and the review domain (Chapter 5.2.2).

Maltparser

Maltparser[12] (Nivre et al., 2006) is a data-driven dependency parser developed at Växjö University and Uppsala University. While a traditional parser-generator constructs a parser given a grammar, a data-driven parser-generator constructs a parser given a Treebank (Marcus, Marcinkiewicz, & Santorini, 1993). MaltParser is an implementation of inductive dependency parsing, where the syntactic analysis of a sentence amounts to the derivation of a dependency structure, and where inductive machine learning is used to guide the parser at nondeterministic choice points.

In this thesis, the dependency representations provided by the version 1.7.2 of Maltparser have been employed as some of the syntactic features used in the negation and speculation detection system for the review domain (Chapter 5.2.2).

[11] See *http://nlp.stanford.edu/software/tagger.shtml*
[12] See *http://www.maltparser.org/*

Appendix E

Publications related to the thesis

Journal papers

- Authors: **Noa P. Cruz**, Manuel J. Maña, Jacinto Mata, Victoria Pachón.
  Title: A Machine Learning Approach to Negation and Speculation Detection in Clinical Texts.
  Journal: Journal of the American Society for Information Science and Technology (JASIST), 2012, vol. 63, no 7, p. 1398-1410.

  Impact factor: 2.005

  Position 20 of 132 in the category Computer Science, Information Systems

- Authors: **Noa P. Cruz**, Manuel J. Maña
  Title: The tokenization problem in the biomedical domain: A comparative study of tools
  Journal: Bioinformatics.
  - Under review

  Impact factor: 5.323

  Position 2 of 47 in the category Mathematical and Computational Biology

- Authors: **Noa P. Cruz**, Maite Taboada, Ruslan Mitkov.
  Title: A Machine-Learning Approach to Negation and Speculation Detection for Sentiment Analysis.
  Journal: Journal of the American Society for Information Science and Technology.
  - Under review

Impact factor: 2.005

Position 20 of 132 in the category Computer Science, Information Systems

Conference papers

- Authors: **Noa P. Cruz Díaz**.
  Title: Detecting Negated and Uncertain Information in Biomedical and Review Texts.
  Conference: Recent Advances in Natural Language Processing (Student Research Workshop).
  Place: Hissar, Bulgaria.
  Date: September, 2013.

- Authors: **Noa P. Cruz Díaz**.
  Title: Negation and Speculation Detection for Improving Information Retrieval Effectiveness.
  Conference: Fifth BCS-IRSG Symposium on Future Directions in Information Access (FDIA 2013).
  Place: Granada, Spain.
  Date: September, 2013.

- Authors: Natalia Konstantinova, Sheila C.M. de Sousa, **Noa P. Cruz**, Manuel J. Maña, Maite Taboada, Ruslan Mitkov.
  Title: A review corpus annotated for negation, speculation and their scope.
  Conference: The International Conference on Language Resources (LREC).
  Place: Istanbul, Turkey.
  Date: May, 2012.

- Authors: J.M. Córdoba, M.J. Maña, **N.P. Cruz**, J. Mata, F. Aparicio, M. Buenaga, D. Glez-Peña, F. Fdez-Riverola.
  Title: Medical-Miner at TREC 2011 Medical Records Track.
  Conference: Text REtrieval Conference – TREC Medical Records Track.
  Place: Gaithersburg, Md. (EE.UU.).
  Date: November, 2011.

- Authors: **Noa P. Cruz**, Manuel J. Maña y Jacinto Mata.
  Title: Aprendizaje Automático versus Expresiones Regulares en la Detección de la Negación y la Especulación en Biomedicina.
  Conference: XXVI Congreso de la Sociedad Española de Procesamiento del Lenguaje Natural.
  Place: Valencia, Spain.
  Date: September, 2010.

- Authors: Jacinto Mata, Manuel J. Maña, José M. Bermúdez, **Noa P. Cruz**, Patricia Jiménez.
  Title: Handling Negation in Classification of Clinical Texts.
  Conference: AMIA Workshop on Challenges in Natural Language Processing for Clinical Data.
  Place: Washington, D.C. (EE.UU.).
  Date: November, 2008.

Bibliography

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

Agarwal, S., & Yu, H. (2010b). Biomedical negation scope detection with conditional random fields. *Journal of the American Medical Informatics Association : JAMIA, 17*(6), 696-701. doi:10.1136/jamia.2010.003228; 10.1136/jamia.2010.003228.

Akbani, R., Kwek, S., & Japkowicz, N. (2004). Applying support vector machines to imbalanced datasets. *Machine learning: ECML 2004* (pp. 39-50) Springer.

Ananiadou, S., Kell, D. B., & Tsujii, J. (2006). Text mining and its potential applications in systems biology. *Trends in Biotechnology, 24*(12), 571-579.

Androutsopoulos, I., & Malakasiotis, P. (2009). A survey of paraphrasing and textual entailment methods. *arXiv Preprint arXiv:0912.3747.*

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,* 283-287.

Averbuch, M., Karson, T., Ben-Ami, B., Maimon, O., & Rokach, L. (2004). Context-sensitive medical information retrieval. *Proc. of the 11th World Congress on Medical Informatics (MEDINFO-2004),* 1-8.

Baker, K., Bloodgood, M., Dorr, B. J., Filardo, N. W., Levin, L. S., & Piatko, C. D. (2010). A modality lexicon and its use in automatic tagging. *LREC.*

Ballesteros Martínez, M. (2010). *Mejora De La Precisión Para El Análisis De Dependencias Usando Maltparser Para El Castellano.*

Ballesteros, M., Francisco, V., Díaz, A., Herrera, J., & Gervás, P. (2012). Inferring the scope of negation in biomedical documents. *Computational linguistics and intelligent text processing* (pp. 363-375) Springer.

Barrett, N. (2012). *Natural Language Processing Techniques for the Purpose of Sentinel Event Information Extraction.*

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

Beesley, K. R., & Karttunen, L. (2003). Finite-state morphology: Xerox tools and techniques. *CSLI, Stanford.*

Benamara, F., Chardon, B., Mathieu, Y. Y., Popescu, V., & Asher, N. (2012). *How do negation and modality impact opinions?*

Bird, S., Klein, E., & Loper, E. (2009). *Natural language processing with python* O'Reilly Media, Inc.

Blanco, E., & Moldovan, D. (2011a). Semantic representation of negation using focus detection. *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1,* 581-589.

Blanco, E., & Moldovan, D. I. (2011b). Some issues on detecting negation from text. *FLAIRS Conference.*

Brill, E. (1992). A simple rule-based part of speech tagger. *Proceedings of the Workshop on Speech and Natural Language,* 112-116.

Browne, A. C., Divita, G., Aronson, A. R., & McCray, A. T. (2003). UMLS language and vocabulary tools. *AMIA Annual Symposium Proceedings,* 798.

Campbell, D. A., & Johnson, S. B. (2001). Comparing syntactic complexity in medical and non-medical corpora. *AMIA Annual Symposium Proceedings*, 90-94.

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

Carpenter, B., & Baldwin, B. (2011). Natural language processing with LingPipe 4.

Carreras, X., Chao, I., Padró, L., & Padró, M. (2004). FreeLing: An open-source suite of language analyzers. *LREC.*

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.

Chen, H., & Sharp, B. M. (2004). Content-rich biological network constructed by mining PubMed abstracts. *BMC Bioinformatics, 5*(1), 147.

Clegg, A. B. (2008). *Computational-Linguistic Approaches to Biological Text Mining.*

Clegg, A. B., & Shepherd, A. J. (2007). Benchmarking natural-language parsers for biological applications using dependency graphs. *BMC Bioinformatics, 8*(1), 24.

Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and Psychological Measurement, 20*(1), 37-46. doi:10.1177/001316446002000104.

Collier, N., Park, H. S., Ogata, N., Tateishi, Y., Nobata, C., Ohta, T., . . . Tsujii, J. (1999). The GENIA project: Corpus-based knowledge acquisition and information extraction from genome research papers. *Proceedings of the Ninth Conference on European Chapter of the Association for Computational Linguistics,* 271-272.

Copestake, A. A., & Flickinger, D. (2000). An open source grammar development environment and broad-coverage english grammar using HPSG. *LREC.*

Córdoba, J. M., Maña M. J., Cruz N. P., Mata, J., Aparicio, F., Buenaga, M., Glez-Peña D., Fdez-Riverola F. (2011). Medical-Miner at TREC 2011 Medical Records Track. *Text REtrieval Conference – TREC Medical Records Track*.

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.*

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.

Cruz Díaz, N.P., Maña López, M.J. (2014). The tokenization problem in the biomedical domain: a comparative study of tools. *Information Processing & Management.*

Cruz Díaz, N.P., Taboada, M, Mitkov R. (2014). A Machine-Learning Approach to Negation and Speculation Detection for Sentiment Analysis. *Journal of the American Society for Information Science and Technology.*

Cunningham, H., Maynard, D., Bontcheva, K., & Tablan, V. (2002). GATE: An architecture for development of robust HLT applications. *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics,* 168-175.

Cunningham, H., Tablan, V., Roberts, A., & Bontcheva, K. (2013). Getting more out of biomedical documents with GATE's full lifecycle open source text analytics. *PLoS Computational Biology, 9*(2), e1002854.

Dadvar, M., Hauff, C., & de Jong, F. (2011). Scope of negation detection in sentiment analysis.

de Buenaga, M., Fdez-Riverola, F., Maña, M., Puertas, E., Glez-Peña, D., & Mata, J. (2010). Medical-miner: Integración de conocimiento textual explícito en técnicas de minería de datos para la creación de herramientas traslacionales en medicina. *Procesamiento Del Lenguaje Natural, 45*, 319-320.

De Haan, F. (1997). *The interaction of modality and negation: A typological study* Taylor & Francis.

de Marneffe, M., MacCartney, B., Grenager, T., Cer, D., Rafferty, A., & Manning, C. D. (2006). Learning to distinguish valid textual entailments. *Second Pascal RTE Challenge Workshop.*

Denny, J. C., Miller, R. A., Waitman, L. R., Arrieta, M. A., & Peterson, J. F. (2009). Identifying QT prolongation from ECG impressions using a general-purpose natural language processor. *International Journal of Medical Informatics, 78*, S34-S42.

Denny, J. C., Choma, N. N., Peterson, J. F., Miller, R. A., Bastarache, L., Li, M., & Peterson, N. B. (2012). Natural language processing improves identification of colorectal cancer testing in the electronic medical record. *Medical Decision Making : An International Journal of the Society for Medical Decision Making, 32*(1), 188-197. doi:10.1177/0272989X11400418; 10.1177/0272989X11400418.

Di Marco, C., Kroon, F. W., & Mercer, R. E. (2006). Using hedges to classify citations in scientific articles. *Computing attitude and affect in text: Theory and applications* (pp. 247-263) Springer.

Dowty, D. (1994). The role of negative polarity and concord marking in natural language reasoning. *Proceedings of SALT, , 4* 114-144.

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.

Evang, K., Basile, V., Chrupała, G., & Bos, J. (2013). Elephant: Sequence labeling for word and sentence segmentation. *Proceedings of the EMNLP 2013: Conference on Empirical Methods in Natural Language Processing, Seattle, United States.*

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,* 1-12.

Fiszman, M., Rindflesch, T. C., & Kilicoglu, H. (2006). Summarizing drug information in medline citations. *AMIA ...Annual Symposium Proceedings / AMIA Symposium.AMIA Symposium,* , 254-258.

Flickinger, D. (2000). On building a more efficient grammar by exploiting types. *Natural Language Engineering, 6*(1), 15-28.

Friedman, C., Alderson, P. O., Austin, J. H., Cimino, J. J., & Johnson, S. B. (1994). A general natural-language text processor for clinical radiology. *Journal of the American Medical Informatics Association : JAMIA, 1*(2), 161-174.

Fukuda, K., Tsunoda, T., Tamura, A., & Takagi, T. (1998). Toward information extraction: Identifying protein names from biological papers. *Pac Symp Biocomput, , 707*(18) 707-718.

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

Ganter, V., & Strube, M. (2009). Finding hedges by chasing weasels: Hedge detection using wikipedia tags and shallow linguistic features. *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers,* 173-176.

Gantner, F., Schweiger, C., & Schlander, M. (2002). Naming, classification, and trademark selection: Implications for market success of pharmaceutical products. *Drug Information Journal, 36*(4), 807-824.

García, S., Fernández, A., & Herrera, F. (2009). Enhancing the effectiveness and interpretability of decision tree and rule induction classifiers with evolutionary training set selection over imbalanced problems. *Applied Soft Computing, 9*(4), 1304-1314.

Gelbukh, A., Torres, S., & Calvo, H. (2005). Transforming a constituency treebank into a dependency treebank.

Georgescul, M. (2010). A hedgehop over a max-margin framework using hedge cues. *Proceedings of the Fourteenth Conference on Computational Natural Language Learning---Shared Task,* 26-31.

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

Goldin, I., & Chapman, W. W. (2003). Learning to detect negation with 'not'in medical texts. *Proc Workshop on Text Analysis and Search for Bioinformatics, ACM SIGIR.*

Grabar, N., & Hamon, T. (2009). Exploitation of speculation markers to identify the structure of biomedical scientific writing. *AMIA Annual Symposium Proceedings,* 203-207.

Grange, B., & Bloom, D. (2000). Acronyms, abbreviations and initialisms. *BJU International, 86*(1), 1-6.

Grefenstette, G., & Tapanainen, P. (1994). *What is a word, what is a sentence?: Problems of tokenisation* Rank Xerox Research Centre.

Grover, C., Matheson, C., Mikheev, A., & Moens, M. (2000). LT TTT-A flexible tokenisation tool. *LREC.*

Guo, J. (1997). Critical tokenization and its properties. *Computational Linguistics, 23*(4), 569-596.

Habert, B., Adda, G., Adda-Decker, M., de Marëuil, P. B., Ferrari, S., Ferret, O., . . . Paroubek, P. (1998). Towards tokenization evaluation. *Proceedings of LREC, , 98* 427-431.

Harabagiu, S., Hickl, A., & Lacatusu, F. (2006). Negation, contrast and contradiction in text processing. *AAAI, , 6* 755-762.

Harris, Z. S. (2002). The structure of science information. *Journal of Biomedical Informatics, 35*(4), 215-221.

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

He, Y., & Kayaalp, M. (2006). A comparison of 13 tokenizers on MEDLINE. *Bethesda, MD: The Lister Hill National Center for Biomedical Communications.*

He, H., & Ma, Y. (2013). *Imbalanced learning: Foundations, algorithms, and applications* John Wiley & Sons.

Hintikka, J. (2002). Negation in logic and in natural language. *Linguistics and Philosophy, 25*(5-6), 585-600.

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

Horn, L. R., & Kato, Y. (2000). *Negation and polarity: Syntactic and semantic perspectives: Syntactic and semantic perspectives* Oxford University Press.

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.*

Hu, M., & Liu, B. (2004). Mining and summarizing customer reviews. *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining,* 168-177.

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.

Huddleston, R., & Pullum, G. K. (2002). The cambridge grammar of english. *Language.Cambridge: Cambridge University Press,* , 1-23.

Hyland, K. (1995). The author in the text: Hedging scientific writing. *Hong Kong Papers in Linguistics and Language Teaching, 18*, 33-42.

Hyland, K. (1996). Talking to the academy forms of hedging in science research articles. *Written Communication, 13*(2), 251-281.

Hyland, K. (1998). *Hedging in scientific research articles* John Benjamins Publishing.

Iso, I. (2001). IEC 9126-1: Software engineering-product quality-part 1: Quality model. *Geneva, Switzerland: International Organization for Standardization.*

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,* 1827-1830.

Jiang, J., & Zhai, C. (2007). An empirical study of tokenization strategies for biomedical information retrieval. *Information Retrieval, 10*(4-5), 341-363.

Jin, Y., McDonald, R. T., Lerman, K., Mandel, M. A., Carroll, S., Liberman, M. Y., . . . White, P. S. (2006). Automated recognition of malignancy mentions in biomedical literature. *BMC Bioinformatics, 7*, 492. doi:10.1186/1471-2105-7-492.

Jurafsky, D., & James, H. (2000). Speech and language processing an introduction to natural language processing, computational linguistics, and speech.

Kang, N., van Mulligen, E. M., & Kors, J. A. (2011). Comparing and combining chunkers of biomedical text. *Journal of Biomedical Informatics, 44*(2), 354-360.

Kennedy, A., & Inkpen, D. (2006). Sentiment classification of movie and product reviews using contextual valence shifters. *Computational Intelligence, 22*(2), 110-125.

Kilicoglu, H., & Bergler, S. (2008). Recognizing speculative language in biomedical research articles: A linguistically motivated perspective. *BMC Bioinformatics, 9*(Suppl 11), S10.

Kim, J., Ohta, T., Pyysalo, S., Kano, Y., & Tsujii, J. (2009). Overview of BioNLP'09 shared task on event extraction. *Proceedings of the Workshop on Current Trends in Biomedical Natural Language Processing: Shared Task,* 1-9.

Kim, J. D., Ohta, T., Tateisi, Y., & Tsujii, J. (2003). GENIA corpus--semantically annotated corpus for bio-textmining. *Bioinformatics (Oxford, England), 19 Suppl 1*, i180-2.

Klima, E. S. (1964). Negation in english. *The Structure of Language, 245-323.*

Konstantinova, N., & de Sousa, S. C. (2011). Annotating negation and speculation: The case of the review domain.

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.*

Kratzer, A. (1981). The notional category of modality. *Words, Worlds, and Contexts,* , 38-74.

Krauthammer, M., & Nenadic, G. (2004). Term identification in the biomedical literature. *Journal of Biomedical Informatics, 37*(6), 512-526.

Kulick, S., Bies, A., Liberman, M., Mandel, M., McDonald, R., Palmer, M., . . . White, P. (2004). Integrated annotation for biomedical information extraction. *Proc. of the Human Language Technology Conference and the Annual Meeting of the North American Chapter of the Association for Computational Linguistics (HLT/NAACL),* 61-68.

Kumar, M., & Sheshadri, H. (2012). On the classification of imbalanced datasets. *International Journal of Computer Applications, 44*

Laka, I. (2013). Negation in syntax: On the nature of functional categories and projections. *Anuario Del Seminario De Filología Vasca" Julio De Urquijo", 25*(1), 65-136.

Lakoff, G. (1972). Hedges: A study in meaning criteria and the logic of fuzzy concepts. *Papers of the Chicago Linguistic Society, 8*, 183-228.

Landis, J. R., & Koch, G. G. (1977). The measurement of observer agreement for categorical data. *Biometrics, 33*(1), 159-174.

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

Lawler, J. (2010). Negation and negative polarity. *The Cambridge Encyclopedia of the Language Sciences.*

Lease, M., & Charniak, E. (2005). Parsing biomedical literature. *Natural language Processing–IJCNLP 2005* (pp. 58-69) Springer.

Light, M., Qiu, X. Y., & Srinivasan, P. (2004). The language of bioscience: Facts, speculations, and statements in between. *Proceedings of BioLink 2004 Workshop on Linking Biological Literature, Ontologies and Databases: Tools for Users,* 17-24.

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.

Marcus, M. P., Marcinkiewicz, M. A., & Santorini, B. (1993). Building a large annotated corpus of english: The penn treebank. *Computational Linguistics, 19*(2), 313-330.

Markkanen, R., & Schröder, H. (1989). Hedging as a translation problem in scientific texts. *Special Languages: From Human Thinking to Thinking Machines,* , 171-175.

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

McCallum, A. K. (2002). Mallet: A machine learning for language toolkit.

McClosky, D., & Adviser-Charniak, E. (2010). Any domain parsing: Automatic domain adaptation for natural language parsing.

McClosky, D., & Charniak, E. (2008). Self-training for biomedical parsing. *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies: Short Papers,* 101-104.

McDonald, R., & Pereira, F. (2005). Identifying gene and protein mentions in text using conditional random fields. *BMC Bioinformatics, 6*(Suppl 1), S6.

McDonald, R. T., Winters, R. S., Mandel, M., Jin, Y., White, P. S., & Pereira, F. (2004). An entity tagger for recognizing acquired genomic variations in cancer literature. *Bioinformatics (Oxford, England), 20*(17), 3249-3251. doi:10.1093/bioinformatics/bth350.

Medlock, B. (2008). Exploring hedge identification in biomedical literature. *Journal of Biomedical Informatics, 41*(4), 636-654.

Medlock, B., & Briscoe, T. (2007). Weakly supervised learning for hedge classification in scientific literature. *ACL, , 2007* 992-999.

Mierswa, I., Wurst, M., Klinkenberg, R., Scholz, M., & Euler, T. (2006). Yale: Rapid prototyping for complex data mining tasks. *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining,* 935-940.

Mitchell, K. J. (2004). Implementation and evaluation of a negation tagger in a pipeline-based system for information extraction from pathology reports.

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,* 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,* 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,* 21-29.

Morante, R., & Daelemans, W. (2012). ConanDoyle-neg: Annotation of negation in conan doyle stories. *Proceedings of the Eighth International Conference on Language Resources and Evaluation.*

Morante, R., Liekens, A., & Daelemans, W. (2008). Learning the scope of negation in biomedical texts. *Proceedings of the Conference on Empirical Methods in Natural Language Processing,* 715-724.

Morante, R., & Sporleder, C. (2012). Modality and negation: An introduction to the special issue. *Computational Linguistics, 38*(2), 223-260.

Morante, R., Van Asch, V., & Daelemans, W. (2010). Memory-based resolution of in-sentence scopes of hedge cues. *Proceedings of the Fourteenth Conference on Computational Natural Language Learning---Shared Task,* 40-47.

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.

Nawaz, R., Thompson, P., & Ananiadou, S. (2010). Evaluating a meta-knowledge annotation scheme for bio-events. *Proceedings of the Workshop on Negation and Speculation in Natural Language Processing,* 69-77.

Nivre, J., Hall, J., & Nilsson, J. (2006). Maltparser: A data-driven parser-generator for dependency parsing.

Ohta, T., Tateisi, Y., & Kim, J. (2002). The GENIA corpus: An annotated research abstract corpus in molecular biology domain. *Proceedings of the Second International Conference on Human Language Technology Research,* 82-86.

Olivier Bodenreider, B. S., & burgun, A. (2004). The ontology-epistemology divide: A case study in medical terminology. *Formal Ontology in Information Systems: Proceedings of the Third International Conference (FOIS-2004),* 185.

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

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: Volume 3-Volume 3,* 1398-1407.

Padró, L., & Stanilovsky, E. (2012). Freeling 3.0: Towards wider multilinguality.

Palmer, F. R. (2001). *Mood and modality* Cambridge University Press.

Pang, B., & Lee, L. (2004). *A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts.*

Payne, T. E. (1997). *Describing morphosyntax: A guide for field linguists* Cambridge University Press.

Pestian, J. P., Brew, C., Matykiewicz, P., Hovermale, D., Johnson, N., Cohen, K. B., & Duch, W. (2007). A shared task involving multi-label classification of clinical free text. *Proceedings of the Workshop on BioNLP 2007: Biological, Translational, and Clinical Language Processing,* 97-104.

Platt, J. C.12 fast training of support vector machines using sequential minimal optimization.

Prabhakaran, V., Rambow, O., & Diab, M. (2010). Automatic committed belief tagging. *Proceedings of the 23rd International Conference on Computational Linguistics: Posters,* 1014-1022.

Proux, D., Rechenmann, F., Julliard, L., Pillet, V., & Jacq, B. (1998). Detecting gene symbols and names in biological texts: A first step toward pertinent information extraction. *Genome Informatics Series,* , 72-80.

Quinlan, J. R. (1986). Induction of decision trees. *Machine Learning, 1*(1), 81-106.

Quinlan, J. R. (1993). *C4. 5: Programs for machine learning* Morgan kaufmann.

Ratnaparkhi, A. (1996). A maximum entropy model for part-of-speech tagging. *Proceedings of the Conference on Empirical Methods in Natural Language Processing, , 1* 133-142.

Rei, M., & Briscoe, T. (2010). Combining manual rules and supervised learning for hedge cue and scope detection. *Proceedings of the Fourteenth Conference on Computational Natural Language Learning---Shared Task,* 56-63.

Ricardo, B., & Berthier, R. (2011). Modern information retrieval: The concepts and technology behind search second edition. *Addision Wesley.*

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

Rinaldi, F., Schneider, G., Kaljurand, K., Hess, M., & Romacker, M. (2006). An environment for relation mining over richly annotated corpora: The case of GENIA. *BMC Bioinformatics, 7*(Suppl 3), S3.

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.*

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.

Seifert, S., & Welte, W. (1987). *A basic bibliography on negation in natural language* Gunter Narr Verlag.

Shatkay, H., Pan, F., Rzhetsky, A., & Wilbur, W. J. (2008). Multi-dimensional classification of biomedical text: Toward automated, practical provision of high-utility text to diverse users. *Bioinformatics (Oxford, England), 24*(18), 2086-2093. doi:10.1093/bioinformatics/btn381; 10.1093/bioinformatics/btn381.

Smith, L., Rindflesch, T., & Wilbur, W. J. (2004). MedPost: A part-of-speech tagger for bioMedical text. *Bioinformatics, 20*(14), 2320-2321.

Snow, R., Vanderwende, L., & Menezes, A. (2006). Effectively using syntax for recognizing false entailment. *Proceedings of the Main Conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics,* 33-40.

Su, Q., Huang, C., & Chen, H. K. (2010). Evidentiality for text trustworthiness detection. *Proceedings of the 2010 Workshop on NLP and Linguistics: Finding the Common Ground,* 10-17.

Szarvas, G. (2008). Hedge classification in biomedical texts with a weakly supervised selection of keywords. *Proceedings of 46th Meeting of the Association for Computational Linguistics.*

Szarvas, G., Vincze, V., Farkas, R., & Csirik, J. (2008). *The BioScope corpus: Annotation for negation, uncertainty and their scope in biomedical texts.*

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

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

Taboada, M., Voll, K., & Brooke, J. (2008). *Extracting sentiment as a function of discourse structure and topicality.* ().Simon Fraser University.

Tanabe, L., Xie, N., Thom, L. H., Matten, W., & Wilbur, W. J. (2005). GENETAG: A tagged corpus for gene/protein named entity recognition. *BMC Bioinformatics, 6*(Suppl 1), S3.

Tang, B., Wang, X., Wang, X., Yuan, B., & Fan, S. (2010). A cascade method for detecting hedges and their scope in natural language text. *Proceedings of the Fourteenth Conference on Computational Natural Language Learning---Shared Task,* 13-17.

Tateisi, Y., Yakushiji, A., Ohta, T., & Tsujii, J. (2005). Syntax annotation for the GENIA corpus. *Proceedings of the 2nd International Joint Conference on Natural Language Processing (IJCNLP-2005), Jeju Island, Korea, October,* 11-13.

Tomanek, K., Wermter, J., & Hahn, U. (2007a). A reappraisal of sentence and token splitting for life sciences documents. *Medinfo 2007: Proceedings of the 12th World Congress on Health (Medical) Informatics; Building Sustainable Health Systems,* 524.

Tomanek, K., Wermter, J., & Hahn, U. (2007b). Sentence and token splitting based on conditional random fields. *Proceedings of the 10th Conference of the Pacific Association for Computational Linguistics,* 49-57.

Tottie, G. (1991). *Negation in english speech and writing: A study in variation*. New York: Academic Press.

Toutanova, K., Klein, D., Manning, C. D., & Singer, Y. (2003). Feature-rich part-of-speech tagging with a cyclic dependency network. *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1,* 173-180.

Toutanova, K., & Manning, C. D. (2000). Enriching the knowledge sources used in a maximum entropy part-of-speech tagger. *Proceedings of the 2000 Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and very Large Corpora: Held in Conjunction with the 38th Annual Meeting of the Association for Computational Linguistics-Volume 13,* 63-70.

Trieschnigg, D., Kraaij, W., & de Jong, F. (2007). The influence of basic tokenization on biomedical document retrieval. *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval,* 803-804.

Tsuruoka, Y., Tateishi, Y., Kim, J., Ohta, T., McNaught, J., Ananiadou, S., & Tsujii, J. (2005). Developing a robust part-of-speech tagger for biomedical text. *Advances in informatics* (pp. 382-392) Springer.

Tsuruoka, Y., & Tsujii, J. (2005). Bidirectional inference with the easiest-first strategy for tagging sequence data. *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing,* 467-474.

Valencia, V. M. S. (1991). *Studies on natural logic and categorial grammar*.

Van der Wouden, T. (2002). *Negative contexts: Collocation, polarity and multiple negation* Routledge.

Velldal, E., Øvrelid, L., & Oepen, S. (2010). Resolving speculation: MaxEnt cue classification and dependency-based scope rules. *Proceedings of the Fourteenth Conference on Computational Natural Language Learning---Shared Task,* 48-55.

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.

Verbeke, M., Frasconi, P., Van Asch, V., Morante, R., Daelemans, W., & De Raedt, L. (2012). Kernel-based logical and relational learning with kLog for hedge cue detection. *Inductive logic programming* (pp. 347-357) Springer.

Vlachos, A., & Craven, M. (2010). Detecting speculative language using syntactic dependencies and logistic regression. *Proceedings of the Fourteenth Conference on Computational Natural Language Learning---Shared Task,* 18-25.

Von Fintel, K. (2006). Modality and language.

Webster, J. J., & Kit, C. (1992). Tokenization as the initial phase in NLP. *Proceedings of the 14th Conference on Computational Linguistics-Volume 4,* 1106-1110.

Wilson, T., Wiebe, J., & Hoffmann, P. (2005). *Recognizing contextual polarity in phrase-level sentiment analysis.*

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

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,* 714-724.

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,* Seattle, Washington, USA. 968-976.

Zuck, J. G., & Zuck, L. V. (1986). Hedging in newswriting. *Beads Or Bracelets,* , 172-180.