Colección de Monografías de la Sociedad Española para el Procesamiento del Lenguaje Natural (SEPLN). Número 10

Prólogo

La Sociedad Española para el Procesamiento del Lenguaje Natural (SEPLN) es una asociación científica sin ánimo de lucro creada en el año 1983 con el fin de promocionar y difundir todo tipo de actividades relacionadas con la enseñanza, investigación y desarrollo en el campo del procesamiento del lenguaje natural, tanto en el ámbito nacional como internacional.

Entre las actividades principales de la SEPLN figuran:

- La celebración de un congreso anual que sirve de punto de encuentro para los distintos grupos que trabajan en el área del procesamiento del lenguaje natural.
- La edición de una revista científica especializada *Procesamiento del Lenguaje Natural* de periodicidad semestral.
- Un servidor web (www.sepln.org) sobre procesamiento del lenguaje natural donde se encuentran digitalizadas todas las publicaciones de la revista.
- Una lista moderada de correo electrónico (SEPLN-L) que sirve como boletín de información periódica (quincenal) y como espacio de información y discusión para los miembros de la Asociación. La dirección para enviar cualquier comentario o aportación a la lista es sidsepln@si.ehu.es.
- Una convocatoria anual de ayudas para la asistencia al congreso de la SEPLN para jóvenes investigadores.
- Una Edición anual de Premios SEPLN a la Investigación en Procesamiento del Lenguaje Natural.

A esta X Edición de los Premios SEPLN a la Investigación en Procesamiento del Lenguaje Natural se pudieron presentar a concurso trabajos monográficos de investigación originales e inéditos de cualquier extensión, escritos por un miembro de la SEPLN, y que no hubieran sido publicados o enviados a publicación con anterioridad a este concurso. Esta publicación presenta el trabajo premiado este año por la comisión evaluadora.

La Junta Directiva de la SEPLN, en nombre de la Asociación, quiere dejar constancia aquí de la alta calidad de todas las obras presentadas a concurso en esta X Edición de los Premios SEPLN, y animar a todos sus miembros a la participación en sus futuras ediciones. Con la publicación de estas contribuciones en su Colección de Monografías, la SEPLN podrá aportar lo mejor de sus esfuerzos a la actualización y divulgación de la investigación en el campo del procesamiento del lenguaje natural.

Junio 2011
Sociedad Española para el Procesamiento del Lenguaje Natural

Application of Information Extraction techniques to pharmacological domain: Extracting drug-drug interactions

Isabel Segura-Bedmar

Mayo, 2011

En memoria de mi querida amiga María Isabel.
Nunca te olvidaré.

Abstract

A drug-drug interaction occurs when one drug influences the level or activity of another drug. The detection of drug interactions is an important research area in patient safety since these interactions can become very dangerous and increase health care costs. Although there are different databases supporting health care professionals in the detection of drug interactions, this kind of resource is rarely complete. Drug interactions are frequently reported in journals of clinical pharmacology, making medical literature the most effective source for the detection of drug interactions. However, the increasing volume of the literature overwhelms health care professionals trying to keep an up-to-date collection of all reported drug-drug interactions. The development of automatic methods for collecting, maintaining and interpreting this information is crucial for achieving a real improvement in their early detection. Information Extraction (IE) techniques can provide an interesting way of reducing the time spent by health care professionals on reviewing the literature. Nevertheless, no approach has been carried out to extract drug-drug interactions from biomedical texts.

We have conducted a detailed study on various IE techniques applied to biomedical domain. Based on this study, we have proposed two different approximations for the extraction of drug-drug interactions from texts. The first approximation proposes a hybrid approach, which combines shallow parsing and pattern matching to extract relations between drugs from biomedical texts. The second approximation is based on a supervised machine learning approach, in particular, kernel methods. In addition, we have created and annotated the first corpus, DrugDDI, annotated with drug-drug interactions, which allow us to evaluate and compare both approximations. To the best of our knowledge, the DrugDDI corpus is the only available corpus annotated for drug-drug interactions and this research is the first work addressing the problem of extracting drug-drug interactions from biomedical texts. We believe the DrugDDI corpus is an important contribution because it could encourage other research groups to research into this problem. We have also defined three auxiliary processes to provide crucial information, which will be used by the aforementioned approximations. These auxiliary tasks are as follows: (1) a process for text analysis based on the UMLS MetaMap Transfer tool (MMTx) to provide shallow syntactic and semantic information from texts, (2) a process for drug name recognition and classification, and (3) a process for drug anaphora resolution. Finally, we have developed a pipeline prototype which integrates the different auxiliary processes. The pipeline architecture allows us to easily integrate these modules with each of the approaches proposed in this work: pattern-matching or kernels. Several experiments were performed on the DrugDDI corpus. They show that while the first approximation based on pattern matching achieves low performance, the approach based on kernel-methods achieves a performance comparable to those obtained by approaches which carry out a similar task such as the extraction of protein-protein interactions.

Contents

List of Figures

List of Tables

List of Algorithms

List of Examples

Chapter 1

Introduction

1.1. Motivation

During the last years, biomedicine has witnessed a huge development. Large amounts of experimental and computational biomedical data have been generated along with new discoveries, which are accompanied by an exponential increase in the number of biomedical publications describing these discoveries. The continuing growth and diversification of the scientific literature require tremendous systematic and automated efforts to utilize the underlying information. Pharmaceutical industry represents a clear example of the growing biomedical literature. During the whole drug life cycle tens of thousands of documents are generated and must be analyzed. In the near future, tools for knowledge discovery will play a pivotal role in biomedical systems since the overwhelming amount of biomedical knowledge in texts demands automated methods to collect, maintain and interpret them.

Patient safety has become a priority for health systems, since the Institute of Medicine (IOM) of United States, in its *Err is Human* report [Kohn et al., 1999], estimated that between 44 and 98 thousand people die in U.S hospitals each year as the result of problems in patient safety. In recent years, several of the major health organizations such as the World Health Organization (WHO), the Pan American Health Organization (PAHO) and the European Environment and Health Committee (EEHC) have developed strategies to propose plans, actions and legislative measures to control avoidable adverse effects in the most relevant domains of patient safety such as hospital-acquired infections, medications errors, operative and post-operative complications, obstetrics. Some progress has been made in the area of patient safety, however, there is still plenty of room for improvement. Although some areas have effective safety systems, the area of drugs does not appear to have reached the level of development initially expected [Longo et al., 2005, Leape and Berwick, 2005]. Among the advices given for patient safety, major health organizations recommend promoting communication of incidents.

Pharmacovigilance is formally defined by the WHO as "the science and activities related to the detection, assessment, understanding and prevention of adverse effects or any other drug-related problems" [WHO, 2002]. This stage is considered vital by the pharmaceutical companies and other agencies due to the recent high profile safety incidents. Currently, spontaneous post-marketing reporting systems are used when adverse effects are discovered in a drug that is already on the market. Health care professionals are responsible for recognizing and reporting those side effects. Several published drug safety issues have shown that adverse effects of drugs may be detected too late, when millions of patients have already been exposed to them. This fact poses a serious problem for the patient safety, motivating a growing interest for improving these reporting systems. A clear example of it is the *EU-ADR* project for the early detection of adverse drug reactions (ADR) by integrative mining of clinical records and biomedical knowledge.

Following this emphasis, the general aim of this thesis is to improve the early detection of drug-drug interactions (DDIs) in scientific publications, a special case of ADRs. A DDI occurs when one drug influences the level or activity of another, for example, raising its blood levels and possibly intensifying its side effects or decreasing drug concentrations and thereby reducing its effectiveness [Stockley, 2007]. Some DDIs can be beneficial. The pharmacoenhancing effect of ritonavir on lopinavir results in a highly potent, clinically effective antiretroviral drug with a high genetic barrier to viral resistance [von Hentig, 2007]. However, many DDIs can be very dangerous [Aronson, 2007], for example hypoglycaemia can be experienced by patients taking clarithromycin and glibenclamide concurrently.

In Spain, the APEAS [2008] study showed that 47.8% of adverse events are due to drugs, of which 3.5%

result from drug interactions. DDIs can range in severity, including prolonged morbidity and even death. The estimated incidence of DDIs that have a clinical significance ranges from 3% to 20%, depending on how many drugs are taken [Nies, 2001]. The frequency of drug interaction increases disproportionally with the increase in the number of drugs in combination. For example, only 5% of patients with less than six drugs manifested clinical signs of drug interaction; while 40% of patients given 16 drugs experienced an adverse DDI [Naguib et al., 1997]. In addition, DDIs can greatly increase health care costs. A study revealed an increased length of stay of 3.1 days for patients who received warfarin with potentially interacting drugs compared with a control group [Jankel et al., 1994]. Another study [Rosholm et al., 1998] showed that 1.2% of hospital admissions were related to DDIs.

DDIs are a serious problem for patient safety. However, the management of DDIs is a critical issue due to the overwhelming amount of information available on them [Hansten, 2003]. The introduction of new technologies in primary care and hospitals has led to the development of electronic medical record systems, which has opened the possibility of incorporating decision support systems to prevent drug-drug interactions and inform on possible actions to take. However, the deployment of these systems is not widespread yet [Rodríguez-Terol et al., 2009]. The assisted electronic prescription is only available in the 22.4% of the hospitals [Vicedo and Conde, 2007]. In primary care, these systems do not support the management of DDIs.

Therefore, clinicians and pharmacists must be able to manage by themselves the richness of information available on DDIs. There is a great amount of DDI databases [Rodríguez-Terol et al., 2009]. Some of them are Bot-plus [Plus, 2008], Stockley [Stockley, 2007], Drug Interactions Facts [Tatro, 2003]. The diversity of DDI databases currently available poses a significant problem to health care professionals when collecting and evaluating information about a particular interaction from these databases.

Several studies [LAM et al., 2003, Edward et al., 2004] have shown that the quality of the DDI databases is very uneven and the consistency of their content is scarce, so it is very difficult to assign a real clinical significance to each drug interaction. Ideally, prescribing information about a drug should list its potential interactions, together with the following information about each interaction: its mechanism, its relation to the doses of both drugs, its time course, the factors that alter an individual's susceptibility to it, its seriousness and severity, and the probability of its occurrence [Ferner and Aronson, 2006, Aronson, 2004b]. In practice, however, this information is rarely available [Aronson, 2007]. Most DDIs are documented as anecdotal reports or as effects in small studies, in which interactions may be missed if they are limited to a susceptible subset of the population [Aronson, 2004a]. Rodríguez-Terol et al. [2009] established a set of criteria to evaluate and compare 24 databases. The minimum quality criteria includes levels of severity and evidence, bibliographic reference, and the description of clinical management for each drug interaction. They concluded that only 9 databases satisfied the minimum criteria. In particular, an increasingly important issue is the update periods of these databases. The update period is described only in 12 of the 24 databases, from immediate updates to a period of 3 years. Updates over 1 year should be inadmissible, and even more frequent updates should be required.

On the other hand, despite the availability of these databases, a great amount of the most current and valuable information is unstructured, written in natural language and hidden in published articles, scientific journals, books and technical reports. Drug interactions are bread and butter to journals of clinical pharmacology due to the vast number of interactions that can happen [Aronson, 2007]. Each year 300,000 articles are published just within the pharmacology domain [Duda et al., 2005]. Therefore, the medical literature is probably by far the most effective system for detection of DDIs [Aronson, 2007].

The great amount of DDI databases and the deluge of published research have overwhelmed most health care professionals because it is not possible to be kept up-to-date of everything published about drug-drug interactions. Information extraction (IE) from both structured and unstructured data sources can be of great benefit in the pharmaceutical industry allowing identification and extraction of relevant information and providing an interesting way of reducing the time spent by health care professionals on reviewing the literature. In addition, the development of tools for automatically extracting DDIs from biomedical texts is essential for improving and updating the drug knowledge databases.

In this document, we have proposed and evaluated different IE techniques for automatic detection of DDIs from unstructured texts. In particular, we have developed a prototype that allows to combine and compare these techniques. Our approach divides the problem in subtasks such as recognition and classification of pharmacological substances and the detection of interactions between them.

1.2.   Objectives

Due to the dynamic nature of biomedicine, new terms and relations between these terms are constantly rising in the biomedical texts [Ananiadou and McNaught, 2006]. Gene or protein interactions have attracted much attention in the field of biomedical IE and plenty of approaches have been proposed for their detection, however, the automatic extraction of DDIs has been hardly addressed. DDIs have been only tackled by [Duda et al., 2005, Guo and Ramakrishnan, 2009]. Indeed, theses approaches classified and located articles about DDIs but did not mine these interactions explicitly. The solutions for biomedical relation extraction range from simple statistical methods relying on co-occurrences of genes or proteins, methods employing a deep syntactic or semantic analysis, to machine learning techniques. Pattern-based approaches have been widely applied but with limited success because they are not able to correctly process anything other than short and straightforward sentences. Supervised machine learning methods have shown promising results but they have not often been applied in the biomedical domain mainly due to the shortage of gold standard corpora for training and testing these systems. Relations that can span several sentences are rarely tackled. Very few approaches have focused on constructions such as mood, modality and negation, which can significantly alter or even reverse the meaning of the sentence.

The development of automatic methods to produce structured information from unstructured text sources would be extremely valuable to the biomedical community. A structured resource would allow researches and healthcare professionals to write a single query to retrieve all the transcription interactions of any drug. Instead of the thousands of abstract provided by querying the unstructured corpus, the query on the structured corpus might result in a few hundred well formed results; this would obviously save a tremendous amount of time and energy. On the other hand, the extraction of drug-drug interactions can also help to improve the curation process of the drug interactions databases. Most biomedical and pharmacological knowledge resources are manually updated by experts. This manual process is an expensive and laborious task. Our long term goal is to enrich DDI databases by extracting information from large text collections such as MedLine or EMBASE, so we are trying to reduce the manual review work needed. Both are bibliographic databases which contain millions of references to articles published in life science journals. In total, there are 23 million biomedical and pharmacological records from both databases.

The main objective of this thesis is to propose and evaluate information extraction techniques in biomedical documents, particularly, for automatic detection of DDIs from unstructured texts. Our proposal divides the problem in subtasks such as text analysis, recognition and classification of pharmacological substances, resolving drug anaphora and the detection of interactions between drugs. These tasks are crucial to support the extraction of DDIs. The specific objectives addressed in this thesis are listed as follows:

1. The creation of an annotated corpus of drug-drug interactions to evaluate the results of our different experiments. To the best of our knowledge, this corpus is the first corpus for DDIs.

2. Study the main approaches for biomedical IE.

3. The integration of several biomedical knowledge resources into a framework to provide broad coverage for a huge amount of biomedical terms and their relations.

4. Evaluate the usefulness of nomenclature standards in the detection and classification of drugs. The family to which a drug belongs can be an essential clue to automatically detect information regarding its interactions.

5. Develop a framework that allows the study and combination of different IE techniques.

6. Propose a method to resolve the anaphoric expressions involving drugs.

7. Combine the resolution of complex syntactic constructions and a set of patterns defined by a pharmacist in order to extract DDIs.

8. Study the performance of a supervised machine learning method to detect DDIs.

9. Compare both previous approaches and analyze the results.

1.3.   Outline of thesis proposal

This section summarizes our proposal to extract DDIs from biomedical texts. Since the major goal of this thesis is to propose and evaluate IE techniques for automatic detection of DDIs from biomedical texts, we have developed a prototype, which allows us to develop, combine and compare these techniques. The prototype has a modular pipeline architecture, which consists of four main processes shown in figure 1.1. The following subsections briefly describe each process.
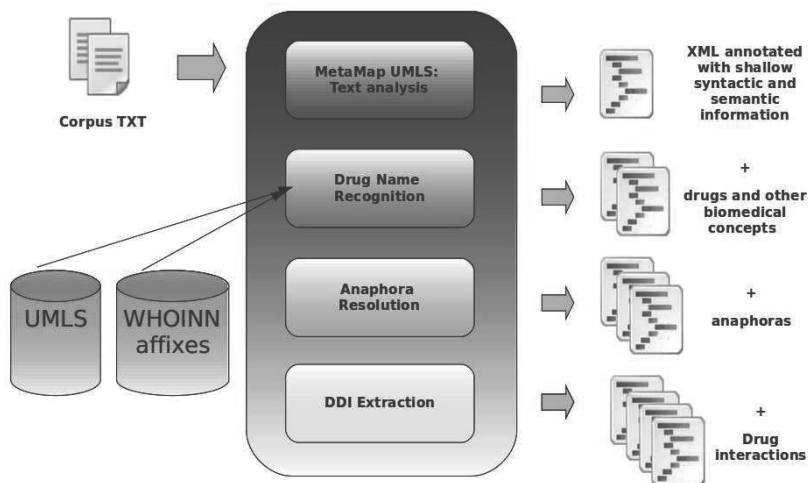
Figure 1.1: Architecture of the prototype

1.3.1.   Text Analysis

This phase is devoted to preprocess the text including the following tasks: tokenization, morphological analysis, PoS tagging, shallow syntactic parsing and semantic annotation. In particular, the UMLS MetaMap Transfer tool (MMTx) [Aronson, 2001] is used to syntactically and semantically analyze these texts.

In addition, the text analysis process is also used in the construction of an annotated corpus with DDIs. While several corpora [Bunescu et al., 2005, Ding et al., 2002, Kim et al., 2003, Nedellec, 2005, Pyysalo et al., 2007] contain annotations of biological relationships such as protein-protein or genetic interactions, there is no corpus annotated with DDIs. For this reason, our first task is to build an annotated corpus for drug-drug interaction extraction. The pharmacological database DrugBank [Wishart et al., 2006, 2007a] offers a complete collection of text documents describing DDIs, which was compiled from several resources and checked by accredited pharmacists. We have used these text documents as a source of unstructured textual information on DDIs. MMTx assists the human annotators by providing lexical, syntactic and semantic information such as sentence boundaries or UMLS concepts. To the best of our knowledge, this corpus is the first gold standard data available for DDI extraction. This corpus allows to evaluate the different experiments. Different approximations can be rated on how close to the gold standard their output is.

1.3.2.   Drug name recognition

In this work, we do not assume that the drugs are given, as it is often the case in other relation extraction works. Instead the recognition of the drug names is solved as a previous task. While most studies of biomedical named entity recognition have mainly focused on genes and proteins, the drug names have not widely addressed. Although many drug information resources are available, and can be used to recognize drug names, there are several open issues such as ambiguous names, synonyms, variations and newly published names related to the drug name recognition, which require special attention.

This thesis proposes a hybrid method which combines information obtained by the MMTx program and nomenclature rules recommended by the WHO International Nonproprietary Names (INN) Programme[WHO, 2006] to identify and classify drug names. Thus, a module called DrugNer has been developed to identify and classify drug names.

1.3.3.   Anaphora resolution

Extracting DDIs from text should take into account the resolution of pronominal and nominal references to entities since interactions are often specified through these references. IE at the sentence level has a limited effect because there are frequent references to previous entities in the discourse, a phenomenon known as 'anaphora'. DDI Extraction is a difficult task whose complexity increases when one or both drugs involved in an interaction are expressed with an anaphoric expression. We have developed two different approaches to address the problem of co-referring expressions in pharmacological literature. In addition, a corpus is developed in order to analyze the phenomena and evaluate both approaches. The first approach is based on a scoring method similar to other works in the biomedical domain [Castano et al., 2002, Lin et al., 2004, Kim and Park, 2004a]. It uses a combination of the domain-specific syntax and semantic information provided by MMTx with generic heuristics. Our second approach uses a set of linguistic rules inspired by Centering Theory Grosz et al. [1995] and constraints satisfaction over the analysis provided by the biomedical syntactic parser MMTx.

1.3.4.   Extraction of Drug-Drug Interactions

In this thesis, we propose two approximations for the extraction of drug-drug interactions from biomedical texts. The first approximation is a hybrid approach which combines shallow parsing and pattern matching to extract relations between drugs from biomedical texts. Our second approximation is based on the kernel methods presented by Giuliano et al. [2006].

We should note that most of the existing approaches for relation extraction usually assume that the argument entities of the relation occur in the the same sentence [Sarawagi, 2007]. We also assume the scope of the interaction is the sentence, that is, interactions which span several sentences are not addressed by neither of our approximations.

1.4.   Document Structure

As it was pointed out in the previous section, this thesis proposes two different approximations for the automatic extraction of DDIs from texts. In addition, we also propose a set of complementary techniques to deal with the following tasks: text analysis, drug name recognition and drug anaphora resolution.

The layout of the thesis is split into two main parts. The first one carries out the description of the complementary techniques and the process of construction of an annotated corpus with drug-drug interactions. Thus, each chapter focuses on a particular technique, reviewing the main approaches to this task and describing our particular proposal.

Chapter 2 reviews the methodologies evaluation for Information Extraction in the biomedical domain and the most commonly used measures for evaluating the performance of these systems.

Chapter 3 reviews the main biomedical corpora used in relation extraction task, and describes the process of construction and annotation of the first annotated corpus of drug-drug interactions, the DrugDDI corpus.

Chapter 4 introduces the main approaches to biomedical named entity recognition. Then we propose a method for drug name recognition and classification.

Chapter 5 reviews the main methods for anaphora resolution in the biomedical domain and proposes two different approaches for drug anaphora resolution.

The second part begins with a detailed review of the main approaches to relation extraction in biomedical domain, continues describing the two approximations proposed in this thesis and finishes with some conclusions and future work.

Chapter 6 discusses the state of art in biomedical relation extraction. The problem is analyzed from the perspective of the type of approach. In this chapter, different strategies as well as architectures are described.

A hybrid approach that combines shallow parsing and pattern matching to extract relations between drugs from biomedical texts is presented in chapter 7.

Chapter 8 presents the second approach to the extraction of DDIs, a kernel-based method. We describe the experimental approach followed in this work and compare the performance of both approaches addressed in this thesis.

Finally, Chapter 9 highlights the main conclusions of the work and proposes the future work as a direction for further related research.

Chapter 2

Evaluation of Biomedical Information Extraction Systems

This chapter reviews the main methodologies for IE in the biomedical domain, as well as the most commonly used measures for evaluating the performance of information extraction systems.

The goal of biomedical IE systems is to help biomedical researchers to extract knowledge from the biomedical literature and facilitate new discovery in a more efficient manner. However, such systems are still research prototypes. This is due mainly to the lack of systematic and rigorous evaluation.

2.1. Methodologies

System-oriented and user-oriented paradigms are the main evaluation approaches. While the system-oriented evaluation focuses on the system and/or its components, the user-oriented evaluation focuses on how the system faces to the real world.

Most research in biomedical IE is still devoted to the development of specific functions or algorithms and consequently researchers have chosen to focus on the evaluation of specific components of systems, such as named entity recognition or detection of relationships instead of in a complete system such as medical question answering application. However, until recently, it has not been possible to compare different approaches, because the various groups involved were addressing different problems and often using private datasets. In the absence of shared datasets and standardized evaluation measures, it was no possible to compare them. In the last years, several workshops on biomedical text processing and text mining for biology have been held. These workshops have made it possible for different approaches can be compared with each other through the introduction of common evaluation frameworks, shared resources, and standardized metrics. This allowed the research community to assess what techniques do and do not work, and to demonstrate the progress being made in these fields. Some of the challenge evaluations to date for biomedical IE are listed below:

- Knowledge Discovery and Data Mining (KDD) Challenge Cup 2002 [Yeh et al., 2002] focused on constructing models to assist genome annotators by automatically extracting information from scientific articles.

- TREC Genomics Track [Hersh et al., 2005]: Document retrieval and classification tasks for genomics.

- Critical Assessment of Information Extraction in Biology (BioCreAtIvE)
  [Hirschman et al., 2005, Krallinger et al., 2009] focuses on two tasks: gene mention identification and normalization and extraction of protein-protein interactions (PPI) from text.

- DTMBIO [Song et al., 2009] and BioNLP [Demner-Fushman et al., 2008]: Both workshops cover a wide range of topics from most areas of natural language processing and from both the clinical and the genomics domains.

However, these assessments are still very limited in discerning the larger role of IE as a tool for real-world biomedical researchers. Thereby, it would be desirable to undertake user-oriented evaluations to determine the most effective use of these systems for their intended audience.

2.2. Evaluation Measures

It is critical to select clear, reproducible, and easily understood evaluation metrics. To introduce the metrics, it is necessary to define a structure known as a *confusion matrix* or *contingency table*. The confusion matrix has four categories:

- *True Positives (TP)* are examples correctly labeled as positives.

- *False Positives (FP)* refer to negative examples incorrectly labeled as positive.

- *True negatives (TN)* correspond to negatives correctly labeled as negative.

- *False negatives (FN)* refer to positive examples incorrectly labeled as negative.

	actual positive	actual negative
predicted positive	TP	FP
predicted negative	FN	TN

Table 2.1: Confusion Matrix

True positives and false negatives, as well as false positives and true negatives are mutually related to. We define N^+ as the total number of positive examples, and N^- as the total number of negative examples. Thus, it is clear that TP and FN are the complementary labels for N^+, and similarly TN and FP for N^-. TP and FP are the only two of the four numbers TP, FP, TN, FN that are independent.

$$N^+ = TP + FN \tag{2.1}$$

$$N^- = TN + FP \tag{2.2}$$

The following list defines various metrics based on the confusion matrix:

- The *True Positive Rate (TPR)* measures the fraction of positive examples that are correctly labeled. TPR has also been referred to as *sensitivity* or *recall*. This score is usually used for evaluating the performance of medical tests.

$$TPR = Sensitivity = Recall = \frac{TP}{TP + FN} = \frac{TP}{N^+} \tag{2.3}$$

- The *False Positive Rate (FPR)* measures the fraction of negative examples that are misclassified as positive.

$$FPR = \frac{FP}{TN + FP} = \frac{FP}{N^-} \tag{2.4}$$

- *True Negative Rate (TNR)* as measures the fraction of negative examples that are correctly labeled. TNR has also been referred to as *specificity*. In medical applications, FPR is replaced with TNR.

$$TNR = Specificity = \frac{TN}{TN + FP} = \frac{TN}{N^-} \tag{2.5}$$

- *False Negative Rate (FNR)* as measures the fraction of positive examples that are misclassified as negative.

$$FNR = \frac{FN}{FN + TP} = \frac{FN}{N^+} \tag{2.6}$$

Then, it is clear that:

$$TPR + FNR = 1 \tag{2.7}$$

$$FPR + TNR = 1 \tag{2.8}$$

To evaluate the performance of an IE system, normally recall and precision values are measured. Recall has been previously defined (equation 2.3). Precision is the proportion of detected examples that are actually positive examples:

$$Precision = \frac{TP}{TP + FP} \qquad (2.9)$$

An ideal information extracting system should fulfill $FN = 0$, $FP = 0$. Precision and recall stand in opposition to one another. As precision goes up, recall usually goes down (and vice versa). *F-measure* is defined by the harmonic (weighted) average of precision and recall where the parameter *beta* indicates a relative weight of precision with respect to the recall. With $\beta = 1$, the balanced *F-score* (F_1) is obtained in which recall and precision are evenly weighted; when $\beta = 2$ (F_2), an overall performance but giving more importance to recall (twice as much as precision) is achieved.

$$F(\beta) = \frac{(1 + \beta^2)PR}{\beta^2 P + R} \qquad (2.10)$$

2.3. Unsolved issued in evaluation process

If the promise of IE techniques to enhance biomedical research is to be met, better evaluation is essential. This will not only help the field better determine what approaches work best, but also provide insight into how systems can enhance the work of their intended users.

Different researchers not only use different collections, but also use those that do exist in different ways. A number of researchers have developed their own collections for use by their own research systems. For example, the systems Textpresso [Muller et al., 2004b] and MedScan [Novichkova et al., 2003], have their own test collections that have been used only for their evaluation. Similarly, the well-known Abgene tagger [Tanabe and Wilbur, 2002] has a test collection that has not been re-used by other research groups. A major problem with collections that are not shared is that the results of reported findings cannot be replicated, compared or improved upon.

Therefore, an important challenge is the development of the gold standard for evaluation systems. A gold standard dataset or corpus is one whose annotation has been checked and corrected. This is typically carried out in order to evaluate automatic annotation systems. Different programs can be rated on how close to the gold standard their output is. However, the development of this gold standard is still under way, far from maturity, which requires more concerted efforts. The experience in the newswire domain shows that the construction of evaluation benchmarks in the face of common challenges contribute greatly to the rapid development of IE.

Efforts will be required to focus on linking the knowledge in the databases with text sources available. In the future, biomedical IE might provide new approaches for relation discovery that exploit efficiently indirect relationships derived from bibliographic analysis of entities contained in biological databases. In addition, the developers of IE systems need to improve the test collections for system-oriented evaluation and undertake user-oriented evaluations to determine how systems will be most effective in real world settings.

The main objective of this thesis is to propose and evaluate information extraction techniques in biomedical documents for automatic extraction of DDIs from biomedical texts. Our proposal divides the problem in tasks such as text analysis, recognizing and classification of pharmacological substances, resolving drug anaphora and the detection of interactions between drugs. For this reason, the evaluation methodology followed in this thesis, will focus on those specific tasks rather than on the user needs. In particular, we use the standard metrics precision, recall and f-measure for the evaluation.

Chapter 3

DrugDDI: an annotated corpus for Drug-Drug Interaction Extraction

An annotated corpus is a collection of texts that have been tagged with linguistic information (part-of-speech tags, syntactic structure, co-references, etc) or domain knowledge information (named entities and relations). Annotated corpora are valuable resources as they provide the gold standard data for the systematic automatic evaluation of the NLP techniques. During the last decade, there has been a surge of interest in using NLP techniques to retrieving and extracting information from the biomedical texts. Thus, there is an increased demand for building corpora annotated with biomedical entities and relations.

The goals of this chapter are: (1) to review the biomedical corpora used by NLP techniques for the relation extraction, and (2) to describe the process of construction and annotation of the first annotated corpus of drug-drug interactions, the DrugDDI corpus. Section 3.1 concludes while several corpora contain annotations of biological relationships such as protein-protein or genetic interactions, there is no corpus annotated with DDIs. Therefore, one of the main objectives of the thesis is to provide an annotated corpus for DDI extraction. The creation of the DrugDDI corpus is described in the section 3.2

3.1. Biomedical corpora for relation extraction

During the last years, several biomedical corpora have been developed for evaluating the performance of NLP techniques. Text collections such as Ohsumed [Hersh and Bhupatiraju, 2003], TREC Genomics track data sets [Hersh and Bhupatiraju, 2003, Hersh et al., 2004], or the evaluation dataset for the Protein interaction article subtask [Krallinger and Valencia, 2007] of the second BioCreative challenge, are very useful for the information retrieval research. Several biomedical annotated corpora are available for named entity recognition. Some of these corpora are Genia [Ohta et al., 2002, Kim et al., 2003], Yapex [Franzen et al., 2002], GENETAG [Tanabe et al., 2005] and MedStract [Pustejovsky et al., 2002b]. They have been annotated with semantic classes relevant to the molecular biology domain such as gene, protein, or cell, among others. MedStract is also annotated for anaphora resolution. The size of these corpora is relatively small, not exceeding 2,000 abstracts. A detailed comparative analysis of the above corpora can be found in [Cohen et al., 2005]. Currently, the CALBC project [Rebholz-Schuhmann et al., 2009] (Collaborative Annotation of a Large Biomedical Corpus) aims to build a large annotated corpus made up of 150,000 Medline abstracts by the integration of the annotations provided by several named entity recognition systems. We believe that the CALBC project can make a significant contribution to build large scale annotated corpus, which are so necessary to train text mining systems.

It is not our intention to provide a complete review of all biomedical annotated corpora, instead, we focus on the major biomedical corpora annotated with biomedical relationships. The PDG corpus was automatically constructed using the PPIs extraction system presented in [Blaschke et al., 1999]. Although, the corpus was manually verified, it contains some errors caused by the named entity tagger and the relation extraction module. The corpus consists of 283 sentences and contains 417 PPIs. An important shortcoming of this corpus is that it only contains metadata information about the PPIs such as the list of interaction types, the list of proteins, or the string of text in which the interactions occur, but not any annotation. In addition, this corpus does not contain any linguistic information.

The aim of the Genic Interaction Extraction Challenge (LLL05) [Nedellec, 2005] was to provide a framework for the comparison of the different approaches to learn rules for extracting protein/gene inter-

actions from MedLine abstracts concerning biology. Abstracts were segmented into sentences, selecting only those sentences that contained at least two gene or protein names. Based on the fact that the relevant information is mostly local to single sentences [Ding et al., 2002], the interactions that span several sentences through the use of co-reference were not annotated. Expert biologists annotated the sentences with biological interactions, indicating the roles of the agent and target of the protein/gene names in each interaction. The sentences were also annotated with tokens, lemmas and syntactic dependencies by the use of LinkParser [Sleator and Temperley, 1995] and later manually checked. The corpus was split into training and test sets. The training set includes 80 sentences describing 270 positive examples of genic interactions. The test set includes 87 sentences describing 106 positive examples. The negative examples were not explicitly annotated in the corpus. The data format is similar to that used in the PDG corpus, but the entities and interactions are clearly marked indicating their positions within the text.

The BioInfer corpus [Pyysalo et al., 2007] is made up of 1,100 sentences from Medline abstracts and provided in an XML format. Each sentence contains at least one pair of interacting entities. The sentences were manually annotated with protein, gene and RNA types as well as the interactions between these entities. A relationship ontology was designed in order to classify the interactions. BioInfer captures complex relationships such as n-ary relationships ($n \succeq 2$) or nested relationships. BioInfer is also annotated with syntactic dependencies provided by Link Parser and later manually reviewed. The corpus contains a total of 33,858 tokens, and hence, a relatively high average sentence length, around 30 tokens. The corpus contains a total of 4,573 proteins and 2,662 relationships. The inter-annotator agreement has not been measured.

The AIMed corpus was created by Bunescu et al. [2005], and is considered as the de facto standard for the PPIs extraction task [Airola et al., 2008]. This corpus, made up of 1,000 Medline abstracts, is divided into three datasets which were manually annotated. The first dataset consists of 750 abstracts tagged for gene/protein names containing a total of 5,206 names. The second dataset contains 200 abstracts which were manually annotated with 1,101 PPIs and 4,141 protein names. The third dataset, made up by 30 abstracts, can be used as a source of negative examples as their abstracts do not contain any interaction.

The corpus HPRD50 [Fundel et al., 2007a] consists of 50 Medline abstracts. The biological entities were automatically identified using the ProMiner tagger [Hanisch et al., 2005]. Then, the abstracts were manually annotated by two annotators with biochemical background. The inter-annotator agreement was 81%. The corpus contains a total of 138 protein/gene interactions, corresponding to 92 distinct relations in abstracts.

The corpus IEPA [Ding et al., 2002] contains 303 MedLine abstracts (486 sentences). Each sentence is annotated with PPIs. The corpus contains a total of 335 interactions.

Corpus	Abstracts	Sentences	Relations	Annotations
PDG	–	283	417	PPIs
LLL	–	167	376	entities, PPIs, syntactic dependencies
BioInfer	–	1,100	2,662	entities, PPIs, syntactic dependencies
AIMed (2n dataset annotated with PPIs)	200	–	1,101	entities, PPIs
HPRD50	30	–	138	entities, PPIs
IEPA	303	486	335	PPIs
BioCreAtIvE-PPI	–	1,000	255	entities, PPIs
Li et al. [2008]	240		2,156	entities, biomedical relations
CLEF	150	–	813	entities, biomedical relations

Table 3.1: Comparative analysis among the different biomedical corpora annotated with biomedical relationships.

The BioCreAtIvE-PPI corpus [Krallinger et al., 2008] was built on the GeneTag corpus. One thousand sentences were randomly selected from this corpus and annotated with gene/protein interactions. The corpus contains a total of 255 interactions.

The above corpora focus on the biological domain. BioText [Rosario and Hearst, 2004b] is a corpus for evaluation of mining disease-treatment relations. Unfortunately, this corpus is less useful because it contains some annotations which are not consistent among them.

The CLEF corpus [Roberts et al., 2007, 2009] has been designed to support extracting information from clinical patient reports. This corpus consists of 150 patient records annotated by at least two annotators with 1,161 clinical entity types (for example, *Condition*, *Intervention* or *Drug*) and 813 relation instances (some types of annotated relations are: *has location*, *has indication* or *has finding*). CLEF is also annotated with modifiers, co-references and temporal expressions. The annotators provided a set of guidelines to ensure consistency and also calculated the inter annotator agreement (IAA), achieving a 67% of agreement for entities and 72% for relations.

Li et al. [2008] built a corpus of 200 cancer-related abstracts from Medline, in order to evaluate a kernel method for biomedical relation extraction. A biomedical scientist manually annotated biomedical entities such as genes, proteins, functions, and diseases. The average number of sentences per abstract is 9.08. The corpus consists of 1,815 sentences in the corpus, 1,361 of them contain two or more entities. The average number of entities per sentence is 2.81. The corpus contains a total of 8,071 relation instances, 2,156 out of them are identified as true relations, while the remaining ones are labeled as negative examples.

Recently, Thompson et al. [2009] have built the Gene Regulation Event Corpus (GREC) in which events relating to gene regulation and expression have been annotated by biologist. The arguments of the event verbs are annotated with a semantic role (such as agent, theme, location, temporal, etc). In addition, biomedical concepts are annotated using the Gene Regulation Ontology (GRO) [Beisswanger et al., 2008]. The corpus consists of 240 medline abstracts, containing a total of 3,067 annotated events and 5,026 biological concepts. High levels of agreement are achieved for both the identification of semantic arguments and the assignment of semantic roles to these arguments (88% or above) and for biological category assignment (around 95%).

3.1.1. Open Issues on biomedical corpora for relation extraction

After analyzing previous biomedical corpora, we point some issues concerning relation extraction:

1. Most corpora reviewed here focus on describing the relationships between biological entities, but none contains DDIs.

2. The full articles have not been used to produce this kind of corpus yet. The corpora are made up of abstracts. The average number of sentences per abstract does not exceed 10 sentences. Abstracts are summaries which lost large part of the information contained in their full articles.

3. Corpora are tagged at sentence level and the interactions that span several sentences are not annotated.

4. The complex relationships are only annotated in the BioInfer corpus.

5. BioInfer is the only corpus in which the interactions are classified and mapped to a domain ontology.

6. The size of the different corpora never exceeds 1,000 sentences with an average of 836 interactions per corpus. BioInfer and GREC are the largest corpora with 2,662 relationships and 3,067 events, respectively.

7. The roles of the interacting proteins are only annotated in the LLL corpus.

8. Information about the annotation process is rather scarce. Most of the reviewed corpora do not have (or provide) any information about the inter-annotator agreement nor annotation guidelines.

9. Several corpora have been automatically annotated, and later manually verified. This may mean that they still contain residual errors introduced by the automatic modules.

10. Regarding linguistic information contained within the corpora, LLL, BioInfer and GREC are the only corpora containing syntactic information. Other linguistic phenomena such as negation, temporal information or anaphora have been hardly tackled. Anaphoric expressions and abbreviations have been annotated in the MedStract corpus. BioScope [Szarvas et al., 2008] is a annotated corpus for negations.

11. Structured data files, HTML or XML format have been also used for data representation.

One of the major contributions of this thesis is to construction of the first annotated corpus of DDIs, the DrugDDI corpus. This corpus allow us to automatically evaluate the different approximations proposed in this thesis to extract DDIs. In addition, we think that the corpus can also encourage the NLP

community to research in the pharmacological domain. We have developed the DrugDDI corpus using the XML format. We believe that the use of an standard format will encourage a greater use of this corpus. The corpus contains shallow syntactic and semantic information provided by MMTx. Pharmacological substances as well as other biomedical concepts are automatically annotated. The corpus may also be used to extract other kind of relationships such as drug adverse reaction, food-drug interactions, or drug targets (relation between proteins and drugs). DDIs are manually annotated with the assistance of an expert pharmacist. In addition, the DrugDDI corpus has a considerable size, larger than most in the biomedical corpora.

3.2.   The DrugDDI corpus

The above section shows that there are several annotated corpora with entities and their relations for biomedical domain [Bunescu et al., 2005, Ding et al., 2002, Kim et al., 2003, Nedellec, 2005, Pyysalo et al., 2007], however, DDIs are not included. While the NLP techniques are relatively domain-portable, corpora are not. The lack of an extensively annotated corpus can easily became a bottleneck to apply NLP techniques, specially supervised machine learning algorithms, for extracting DDIs

This section describes the process of the construction and annotation of the DrugDDI corpus. DrugDDI is aiming at providing the gold standard for the application and evaluation of the NLP techniques in the pharmacological domain.

3.2.1.   Collecting the corpus

As source of unstructured textual information on drugs and their interactions, we have used the DrugBank database [Wishart et al., 2006, 2007a]. This database is a rich resource which combines chemical and pharmaceutical information of approximately 4,900 pharmacological substances. DrugBank provides synoptic data about the nomenclature, structure and physical properties of drugs and their drug targets. This type of information is oriented to biochemists and biologists. DrugBank also offers very detailed clinical information about drugs including pharmacology, metabolism and indications. This information is often used by healthcare professionals. Hence, DrugBank has been widely used in several contexts including drug design, drug target discovery or drug interaction prediction, among many other applications. In addition, DrugBank is an online and free resource.

For each drug, DrugBank contains more than 100 data fields such as drug synonyms, brand names, chemical formula and structure, drug categories, ATC and AHFS codes (codes of standard drug families), mechanism of action, indication, dosage forms, toxicity, among other ones. In particular, DrugBank offers a complete collection of DDIs, which was compiled from several resources, checked by an accredited pharmacist and entered manually. This collection consists of 714 food-interactions and 13,242 drug-drug interactions. The interactions are contained in the structured information fields 'Food Interactions' and 'Drug Interactions'.

Additional information can be found in the field 'Interactions'. This field contains a link to a document describing DDIs in unstructured text.This document not only contains a detailed description on the interactions contained in the above structured fields, but also offers information on other interactions which have not collected in them. For example, figure 3.3 shows that *heparin* interacts with drugs such as *doxorubicin*, *droperidol*, *ciprofloxacin*, or *mitoxantrone*. None of them is registered in the above interaction tables. We have used the 'Interactions' field as a source of unstructured textual information on DDIs.

DrugBank provides a file with the names of approved drugs, approximately 1,450. We randomly chose 1,000 drug names and used the RobotMaker[1] application to download the interaction documents for these drugs. The interactions contained in the 'Drug Interactions' tables were also downloaded. These interactions allows us to quantitatively compare the list of automatically DDIs extracted using our system with those contained in an existing resource such as DrugBank. Thus, we can know if our proposed methods are able to detect new interactions that have not been registered in their structured fields yet.

We could only retrieve a total of 930 documents since some drugs do not have any linked document. Due to the annotation process is a very laborious, cost-intensive and time consuming task, we decided to reduce the number of documents to be annotated and only considered 579 documents. We would like to increase the size of annotated corpus. The rest of documents, 352, was dedicated to study the linguistic

---

[1]http://openkapow.com/

Figure 3.1: Drugcard of carpofren in DrugBank.

phenomenon of the pharmaceutical literature. Thus, the corpus is divided into two subsets. Let us name the set of 579 annotated documents as *annotated dataset*, and the rest of documents as *unannotated dataset*.

3.2.2. Processing the corpus

To provide precise description of the text processing stage, we must first introduce the Unified Medical Language System (UMLS) [Bodenreider, 2004, Humphreys et al., 1998]. UMLS is a set of resources developed by the National Library of Medicine (NLM) whose main objective is to assist in the developing of natural language technology for biomedical texts. UMLS has three major knowledge sources: the Metathesaurus, the Semantic Network and the Specialist Lexicon.

Figure 3.2: Drug and Food interactions fields for heparin in DrugBank.

Figure 3.3: Interactions field for heparin in DrugBank

The Metathesaurus is the most comprehensive ontology of the biomedical domain since it integrates a wealth of biomedical terminological resources such as MeSH [Lipscomb, 2000], Diseases Database 2000[2],

---

[2]http://www.diseasesdatabase.com/

SNOMED [Spackman et al., 1997], Gene Ontology [Ashburner et al., 2000], HUGO Gene Nomenclature Database [Povey et al., 2001] and Micromedex DRUGDEX among many others.

All concepts in the Metathesaurus are assigned to at least one semantic type from the UMLS Semantic Network, providing a consistent categorization of all concepts represented in the UMLS Metathesaurus. The Semantic Network contains 135 semantic types such as 'Pharmaceutical substance' (*phsu*), 'Amino Acid, Peptide, or Protein' (*aapp*), 'Disease or Syndrome' (*dsyn*) or 'Gene or Genome' (*gngm*).

Finally, the Specialist Lexicon is a biomedical lexicon with syntactic, morphological and orthographic information.

We use the MMTx tool [Aronson, 2001] to syntactically and semantically analyze the documents in the corpus. The basic function of this program is to map text to concepts in UMLS Metathesaurus. MMTx has been widespread used in IE, Information Retrieval, and Data mining applications [Meystre and Haug, 2006, Díaz-Galiano et al., 2009]. MMTx performs sentence splitting, tokenization, POS-tagging, shallow syntactic parsing, and linking of phrases with UMLS concepts.

Figure 3.5 shows part of the output provided by MMTx for a given document and its transformation to XML format. First, MMTx splits the text into sentences. Second, the phrases in each sentence are identified and classified. MMTx uses the the SPECIALIST minimal commitment parser [McCray et al., 1994] to produce a shallow syntactic parsing of the texts. Table 3.6 shows the different types of phrases that MMTx can identify. If MMTx is not able to determine the type of a given phrase, then MMTx assigns it the label *UNK*, indicating that its type is unknown. For each phrase, it is offered its type, the number of tokens, text and an identifier in the XML document (see figure 3.5).

Then, the parser uses the SPECIALIST lexicon to assign the POS tags to the tokens, and relies on the Xerox part-of-speech tagger [Cutting et al., 1992] when a token has several tags in the lexicon in order to decide the correct tag. Each token is annotated with its POS tag, its word, and a boolean value indicating if it is the head of the phrase (ISHEAD). In addition, the starting and ending offsets of each token within the text are stored in the attributes *start* and *end*, respectively. These character offsets allow to map from the annotation to the raw text easily. For example, figure 3.5 shows the tokens contained in the phrase *with alprazolam,.*
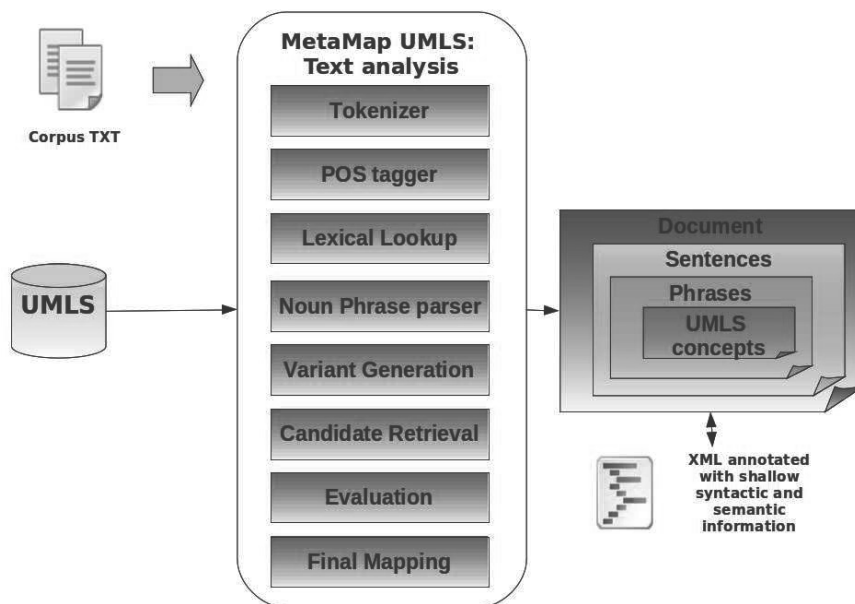
Figure 3.4: MMTx processes.

The XML format provides a maximum flexibility for the use of the DrugDDI corpus. In addition, the corpus is distributed in a *standoff annotation* format that involves storing annotation and text

	annotated dataset	unannotated dataset	total
documents	579	351	930
sentences	5,806	3,255	9,601
phrases	66,021	35,678	101,699
tokens	127,653	69,030	196,683

Table 3.2: Statistics about the DrugDDI corpus

separately [Leech, 1993]. An advantage of the *standoff annotation* format is that the original texts can be immediately retrieved without need of recovering it from the annotations. Besides, this format preserves useful information about the structure of the texts.

Table 3.2 provides information about the number of sentences, phrases and tokens in the DrugDII corpus. As it can be seen from table 3.2.2, the average number of sentences per document (10.3) is greater than in the MedLine abstracts. The average number in the corpus developed in [Li et al., 2008] was 9.08 sentences, while Yu [2006] estimated it in 7.2 ± 1.9 sentences.

Type of Phrase	Annotated Dataset	Unannotated dataset	total
Noun (NP)	18,238	9,982	28,220
Prepositional (PP)	7,607	4,005	11,612
Verbal (VP)	17,271	9,059	26,330
Adjectival (ADJ)	367	176	543
Adverbial (ADV)	2,321	1,332	3,653
Conjunctions (CONJ)	5,772	3,033	8,805
Unknown (UNK)	8,069	4,642	12,711

Table 3.3: Distribution of phrases in DrugDDI

Type of Word	Annotated Dataset	Unannotated dataset	total
adj	10,945	5,963	16,908
adv	4,434	2,534	6,968
aux	5,755	2,877	8,632
compl	523	255	778
conj	5,395	2,814	8,209
det	6,647	3,524	10,171
modal	2,312	1,254	3,566
noun	41,264	22,488	63,752
number	2,507	1,542	4,049
prep	14,129	7,659	21,788
pron	1,164	599	1,763
untagged	47,92	2,247	7,039
verb	8,844	4,746	13,590
mark	18,942	10,528	29,470

Table 3.4: Distribution of tokens in DrugDDI

Element	Avg. per Doc	Avg. per Sentence	Avg. per Phrase
tokens	211.5	20.5	1.9
phrases	109.3	10.6	
sentences	10.3		

Table 3.5: Average number of sentences, phrases and tokens in the DrugDDI corpus.

Table 3.3 shows the distribution of the different types of phrases in the corpus. We can note the *NP*, *VP* and *PP* are the types of phrases that occur more often. We also note that the type *UNK* occurs

```
-<SENTENCE ID="s24" TEXT="Moreover, as noted with alprazolam, the effect of fluvoxamine may even be more
  pronounced when it is administered at higher doses.">
 -<PHRASES>
  +<PHRASE ID="s24.p366" NUMTOKENS="2" TEXT="Moreover" TYPE="UNK"></PHRASE>
  +<PHRASE ID="s24.p367" NUMTOKENS="1" TEXT="as" TYPE="CONJ"></PHRASE>
  +<PHRASE ID="s24.p368" NUMTOKENS="1" TEXT="noted" TYPE="VP"></PHRASE>
  -<PHRASE ID="s24.p369" NUMTOKENS="3" TEXT="with alprazolam" TYPE="PP">
    +<MAPPINGS></MAPPINGS>
    -<TOKENS>
       <TOKEN ISHEAD="false" start="4208" end="4211" ORD="0" POS="prep" WORD="with"/>
       <TOKEN ISHEAD="true" start="4213" end="4222" ORD="0" POS="noun" WORD="alprazolam"/>
       <TOKEN ISHEAD="false" start="4224" end="4224" ORD="2" POS="comma" WORD=","/>
    </TOKENS>
  </PHRASE>
  +<PHRASE ID="s24.p370" NUMTOKENS="2" TEXT="the effect" TYPE="NP"></PHRASE>
  +<PHRASE ID="s24.p371" NUMTOKENS="2" TEXT="of fluvoxamine" TYPE="PP/of"></PHRASE>
  +<PHRASE ID="s24.p372" NUMTOKENS="1" TEXT="may" TYPE="VP"></PHRASE>
  +<PHRASE ID="s24.p373" NUMTOKENS="1" TEXT="even" TYPE="ADV"></PHRASE>
  +<PHRASE ID="s24.p374" NUMTOKENS="1" TEXT="be" TYPE="V/be"></PHRASE>
  +<PHRASE ID="s24.p375" NUMTOKENS="1" TEXT="more" TYPE="ADV"></PHRASE>
  +<PHRASE ID="s24.p376" NUMTOKENS="1" TEXT="pronounced" TYPE="VP"></PHRASE>
  +<PHRASE ID="s24.p377" NUMTOKENS="1" TEXT="when" TYPE="CONJ"></PHRASE>
  +<PHRASE ID="s24.p378" NUMTOKENS="1" TEXT="it" TYPE="NP"></PHRASE>
  +<PHRASE ID="s24.p379" NUMTOKENS="1" TEXT="is" TYPE="V/be"></PHRASE>
  +<PHRASE ID="s24.p380" NUMTOKENS="1" TEXT="administered" TYPE="VP"></PHRASE>
  +<PHRASE ID="s24.p381" NUMTOKENS="4" TEXT="at higher doses" TYPE="PP"></PHRASE>
```

Figure 3.5: Example of a document processed by MMTx.

Type of Phrase	Examples
Noun phrase (NP)	*Drug Interactions, the cytochrome P450 3A4 enzyme system*
Prepositional phrase (PP)	*with drugs, of azole antimycotics, of orally administered midazolam*
Verbal phrase (VP)	*administered,inhibit. decrease*
Adjectival phrase (ADJ)	*hypersensitive*
Adverbial phrase (ADV)	*concurrently, not, to significantly*
Conjunctions (CONJ)	*and,or, since*

Table 3.6: Type of phrases identified by MMTx.

with high frequency. This is due to the documents often contains tables and other kind of enumerations that contain character specials that the parser is not able to identify.

Table 3.4 shows the distribution of the different types of tokens in the corpus. The tag *untagged* is assigned when the tagger is not to able to assign any tag because the token does not exist in the lexicon or is a special token such as a quote, that the tagger is not able to identify. We define the category *mark* to group the punctuation marks and other special characters such as asterisk, dash, braces, binary operators, logic operators, among many others.

Once the shallow syntactic parsing has been performed, MMTx looks for the phrases in the UMLS Metathesaurus. For each phrase, a set of variants is generated using the SPECIALIST lexicon and linguistic techniques to generate its variants. The set of variants consists of the text of the phrase, and its acronyms, abbreviations, synonyms, derivational, inflectional and spelling variants. These variants are looked for in the Metathesaurus, retrieving those concepts that contain at least one of the variants are retrieved. Each concept is evaluated against the text of the phrase using several linguistic metrics to determine its similarity. Finally, those concepts with a highest similarity are selected as the final mapping. For each concept in the final mapping set, MMTx provides its concept unique identifier (CUI), its concept name, and its semantic types. This way, MMTx allows to recognize a variety of biomedical entities occurring in texts. Finally, the output of MMTx is transformed into XML format (see figure 3.5).

Figure 3.6 shows the final mapping retrieved by MMTx for the phrase *Aspirin*. For this phrase, the final mapping consists of an unique concept, *Aspirin*, with CUI *C0004057*, and semantic type *Pharmacological substance*. However, the mapping for a given phrase may consist of a combintation of various candidates involved in disjoint parts of the phrase. For example, in the sentence *Alcohol has a synergistic effect with aspirin in causing gastrointestinal bleeding*, the mappings for the phrase *a synergistic effect* consists of the Metathesaurus concept *Effect* (*C1280500*) and either the concept *Synergism* (*C0599739*), or the concept *Synergist* (*C0920877*).
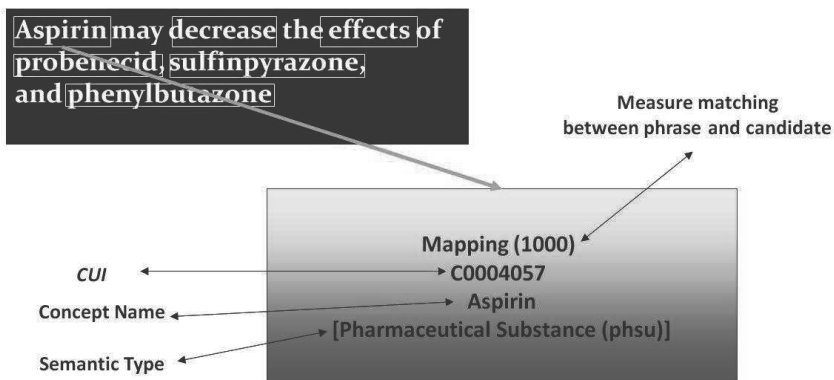
Figure 3.6: Mapping to UMLS concepts for the Aspirin phrase.

3.2.2.1. Failure Analysis of MMTx

Unfortunately, MMTx makes several mistakes. Some mistakes that we detected for the annotation process are described briefly next. However, we omit a deeper discussion about them, since the failure analysis of MMTx is out of scope of this thesis.

MMTx sometimes fails to resolve the correct syntactic type of phrases when they are preceded by a conjunction. Example 1 shows a sentence in which MMTx was not able to classify correctly the phrase (*nefazadone*) in the coordinate structure (the correct type is NP).

Example 1 MMTx fails in classifying the phrase *nefazadone*

$[ketoconazole]_{NP}$, $[ritonavir]_{NP}$, $[nelfinavir]_{NP}$, $[clarithromycin]_{NP}$ and
$[nefazadone]_{UNK}$.

We have also observed a similar situation for the coordinate structures that involve the verbs *to increase* or *to decrease*. These verbs are classified as noun phrases by MMTx when they are part of a coordinate structure (see example 2).

Example 2 MMTx often fails in classifying verb phrases in coordinate structures

Drugs that may either **$[increase]_{NP}$** *or* **$[decrease\ plasma\ phenytoin$
$concentrations]_{NP}$** *include: phenobarbital, vaiproic acid, and sodium valproate.*
Valproic acid has been reported to both **$[increase]_{NP}$** *and* **$[decrease\ ethosuximide\ levels]_{NP}$**.
Coadministration of NIZORAL Tablets and drugs primarily metabolized by the cytochrome P450 3A4 enzyme system may result in increased plasma concentrations of the drugs that could **$[increase_{NP}$** *or prolong both therapeutic and adverse effects.*
The plasma concentration of imipramine may increase when the drug is given concomitantly with hepatic enzyme inhibitors and **$[decrease]_{NP}$** *by concomitant administration of hepatic enzyme inducers.*

MMTx fails to split the phrases when they contain some special character such as '/'. For example, the phrase *No formal drug/drug interaction* is incorrectly segmented into two different phrases: *No formal drug* and */drug interactions.*

MMTx also fails to resolve common abbreviations, for example, the phrase *e.g.* is classified with the following combination of UMLS concepts: *Endosinusial Route of Drug Administration* (*C1522661*) and *APC gene* (*C0162832*). This mistake only happens when the phrase *e.g.* is immediately preceded by a left parenthesis.

Concerning semantic ambiguity, we have observed that some phrases can be linked with several concepts, or even with several combinations of them. MMTx often suggests several senses, but is not able to distinguish the most appropriate one. In the following sentence *This medicine should not be taken with*

MAO inhibitors., the phrase *with MAO inhibitors* is classified by MMTx with the two following UMLS concepts:

1. *Monoamine Oxidase Inhibitors (C0026457)*, a *Pharmacological substance (phsu)*.

2. *MAO Inhibitors - Consent Type (C1548880)*, a *Health Care Activity (hlca)*.

but only the first is the appropriate concept in this context. A reasonable cause can be that the UMLS Metathesaurus contains a large number of ambiguity cases represented by separate concepts, each of which refers to one of the individual senses. However, we believe that the main reason is that the variants of each phrase are generated without taking into account the context information of it.

This problem is even greater when the phrase can be linked with a combination of concepts. For example, MMTx provides seven different combinations of concepts for the phrase *No formal drug interaction studies*, as follows:

1. *Formation* (C1522492) + *Drug Evaluation* (C0013175) + *Drug Interactions* (C0687133)

2. *Manufactured form* (C0376315) + *Drug Evaluation* (C0013175) + *Drug Interactions* (C0687133)

3. *Manufactured form* (C0376315) + *Drug Evaluation* (C0013175) + *Event Qualification - Interaction* (C1546938)

4. *Manufactured form* (C0376315) + *Drug Evaluation* (C0013175) + *Social Interaction* (C0037420)

5. *Qualitative form* (C0348078) + *Drug Evaluation* (C0013175) + *Drug Interactions* (C0687133)

6. *Qualitative form* (C0348078) + *Drug Evaluation* (C0013175) +*Event Qualification - Interaction* (C1546938)

7. *Qualitative form* (C0348078) + *Drug Evaluation* (C0013175) +*Social Interaction* (C0037420)

Regarding the co-reference resolution, phrases such as *this disease*, *the antibiotic*, *this drug* are linked with generic concepts such as *Disease (C0012634)*, *Antibiotics (C0003232)*, or *Pharmaceutical Preparations (C0013227)*, respectively. However, these phrases should be classified into the specific concepts to which they refer in the text.

These are only some of the failures generated by MMTx. A more detailed discussion of the failures of MMTx can be found in the following works [Divita et al., 2004, Bashyam et al., 2007, Meng et al., 2005]. A deeper analysis of them could contribute notably to the improvement of the MMTx tool, but it is out of scope of our objectives in this thesis.

3.2.3.   Annotating the corpus

This section describes the process followed in the annotation of drug and their interactions in the DrugDDI corpus.

As it was seen in the previous subsection, the sentences are analysed by MMTx that provides syntactic and semantic information. In particular, MMTx allows us to recognize and annotate a variety of biomedical entities occurring in texts according to the UMLS semantic types. For example, the figure 3.7 shows the semantic information retrieved by MMTx for the phrases *The anticoagulant effect* and *of heparin*. While the former is classified with the semantic type *Organ or Tissue Function (ortf)* (a physiologic function of a particular organ, organ system, or tissue), *of heparin* is classified with three different semantic types: *Biologically Active Substance (bacs)*, *Carbohydrate (carb)* and *Pharmacological Substance (phsu)*.

The UMLS Semantic Network was reviewed in order to identify the semantic types that can represent drugs useful in this work. Initially, we proposed the following semantic types:

- *Clinical Drug* (*clnd*): A pharmaceutical preparation as produced by the manufacturer. The name usually includes the substance, its strength, and the form, but may include the substance and only one of the other two items. Examples: Zovirax Cold Sore 5% cream, sleeping pill, Acetohexamide 250 MG Oral Tablet.

- *Pharmacological Substance (phsu)*: A substance used in the treatment or prevention of pathologic disorders. This includes substances that occur naturally in the body and are administered therapeutically. Examples: Antiemetics, Cardiovascular Agents, Codeine, Morphine Sulfate.

```
-<SENTENCE ID="s3" TEXT="The anticoagulant effect of heparin is enhanced by concurrent
  treatment with antithrombin III (human) in patients with hereditary antithrombin III deficiency.">
  -<PHRASES>
   -<PHRASE ID="s3.p68" NUMTOKENS="3" TEXT="The anticoagulant effect" TYPE="NP">
     -<MAPPINGS>
        <MAP CUI="C0520996" NAME="Anticoagulant effect" NAME_SHORT="Anticoagulant
        effect" PROB="1000" SEMTYPES="ortf"> </MAP>
      </MAPPINGS>
     +<TOKENS></TOKENS>
     </PHRASE>
   -<PHRASE ID="s3.p69" NUMTOKENS="2" TEXT="of heparin" TYPE="PP/of">
     -<MAPPINGS>
        <MAP CUI="C0019134" NAME="Heparin" NAME_SHORT="Heparin" PROB="1000"
        SEMTYPES="bacs,carb,phsu" STEM_0="-parin" USAN="YES"> </MAP>
      </MAPPINGS>
     +<TOKENS></TOKENS>
     </PHRASE>
```

Figure 3.7: UMLS concepts retrieved by MMTx for the phrases *The anticoagulant effect*, *of heparin* and *of probenecid*

- *Antibiotic (antb)*: A pharmacologically active compound produced by growing microorganisms which kill or inhibit growth of other microorganisms. The direct ancestor of this semantic type is the *phsu* semantic type. Examples: Antibiotics, Cephalosporins, Methicillin.

An experienced pharmacist reviewed the semantic annotation provided by MMTx and recommended us the inclusion of the following UMLS semantic types as possible types of interacting drugs:

- *Biologically Active Substance (bacs)*: A generally endogenous substance produced or required by an organism, of primary interest because of its role in the biologic functioning of the organism that produces it. Examples: Enzyme Precursors, Gastric Acid, Growth Substances.

- *Chemical Viewed Structurally (chvs)*: A chemical or chemicals viewed from the perspective of their structural characteristics. Some examples are Ammonium Compounds, Cations, Siloxanes, Sulfur Compounds.

- *Amino Acid, Peptide, or Protein (aapp)*: Amino acids and chains of amino acids connected by peptide linkages. Examples: Acetylcysteine, Glycoproteins, Peptidyl-Dipeptidase A, glycylglutamine.

Table 3.7 shows the distribution of these semantic types in the corpus. If a term is classified by several drug types, it is only to be counted once. The average number of drugs per document is 24.9, and the average number of drugs per sentence is 2.4.

Semantic Type	Annotated Set	Unannotated set	Total
clnd	106	65	171
phsu	12,767	7,069	1,283,781
antb	695	369	1,064
chvs	60	28	88
bacs	215	110	325
aapp	1,087	619	1,706
Total	14,930	8,260	23,190
Avg. per document	25.8	23.5	24.9
Avg. per sentence	2.6	2.5	2.4

Table 3.7: Distribution of the UMLS semantic types for drugs in the DrugDDI corpus

Drug families (e.g., *analgesics, anticoagulants, salicylates*, etc) are also tagged by MMTx with some of the above semantic types. The automatic extraction of interactions involving drug families is a desirable outcome of our research. If a given interaction involving a drug family is known, then the drugs of this family can be modified by the pharmaceutical industry in order to avoid that interaction. Example 3 contains an interaction involving the drug familiy *Urinary Alkalinizers* and the specific drug *Aspirin*.

Example 3 Drug Family-Drug Interaction

Urinary Alkalinizers decrease aspirin effectiveness by increasing the rate of salicylate renal excretion.

In addition, MMTx allows to classify abstract entities such as *medicine*, *drug*, *medication* as pharmacological substances. The abstract pharmacological substances are usually classified with some of the following concepts: *pharmacological substance* (*C1254351*), or, *pharmaceutical preparations* (*C0013227*). We have decided to preserve this annotation because some interactions involve a specific drug and an anaphoric expression. The recognition of these abstract drugs allows to include in the corpus those interactions that span several sentences by the use of co-reference expressions. For example, the sentence *This medicine can interact with Warfarin* contains two pharmacological substances: a specific drug name *aspirin*, and an abstract drug name *medicine* that refers to a specific drug previously mentioned in the text.

The corpus contains a total of 1,637 terms that map with *pharmacological substance* or *pharmacological preparation* concepts (999 in the annotated dataset). This means that the 7% of all phrases classified as drugs by MMTx, are not physical drugs, but also abstract drugs such as *this drug*, *the medication* or *the medicine*.

The main value of the DrugDDI corpus comes from its annotation since all the documents have been marked-up with drug-drug interactions by a researcher with pharmaceutical background and a pharmacist.

	Annotated Set	Unannotated set	Total
sentences that contain 0 drugs	857	623	1,480
sentences that contain 1 drug	1,175	589	1,764
sentences that contain 2 drugs	1,416	750	2,166
sentences that contain 3 drugs	2,358	1,293	3,651
sentences with at least 2 drugs	3,774	2,043	5,817

Table 3.8: Proportion of drugs in sentences.

Only those documents in the *annotated dataset* have been tagged with drug interactions. Besides, we have only annotated the DDIs at sentence level, that is, those interactions than span several sentences have not been annotated. We have focused on the sentences with at least two drugs.

The *annotated dataset* contains a total of 3,775 sentences than contain two or more drugs (see table 3.8), although only 2,044 contain at least one interaction. A total of 3,160 drug-drug interactions have been identified with the assistance of a pharmacist. Figure 3.8 shows an example of an annotated sentence that contains three interactions. Each interaction is represented as a *DDI* node in which the names of the interacting drugs are registered in its *NAME_DRUG_1* and *NAME_DRUG_2* attributes. The identifiers of the phrases containing these interacting drugs are also annotated. This provides an easily access to the related concepts provided by MMTx.

Figure 3.8: Example of DDI annotations.

Ideally, we should also annotate additional information about each interaction such the roles of each drug, its mechanism, its relation to the doses of both drugs, its time course, the factors that alter an individual's susceptibility to it, its seriousness and severity, and the probability of its occurrence, when these data are available in the text. We are planning to expand the annotations to include this type of data. Furthemore, it would be desirable to classify the DDIs according to a drug knowledge database. Example 4 contains some sentences annotated with interactions. The corpus also contains sentences that can be used to obtain explicit examples of non-interactions (see example 5).

Example 4 Examples of annotated interactions

(1) Interaction between a specific drug and a drug family: *[**Aspirin**]$_{drug}$ is contraindicated in patients who are hypersensitive to [**nonsteroidal anti-inflammatory agents**]$_{drugfamily}$.*
(2) A sentence containing several interactions: *[**Dexbrompheniramine**]$_{drug}$ can interact with [**alcohol**]$_{drug}$ or [**other CNS depressants**]$_{drugfamily[]}$*
(3) A probable interaction (that is, it may not happen): *[**Propantheline**]$_{drug}$ and [**diphenoxylate**]$_{drug}$ may increase [**digoxin**]$_{drug}$ absorption.. This kind of interactions only happen under given conditions.*
(4) An interaction that depends on its drug dosages and its time course: *A literature article reported that when a [**60 mg controlled-release morphine capsule**]$_{drug}$ was administered 2 hours prior to a [**600 mg Neurontin capsule (N=12)**]$_{drug}$, mean gabapentin AUC increased by 44% compared to gabapentin administered without morphine*

Example 5 Negation of DDIs.

*In vitro binding studies with human serum proteins indicate that glipizide binds differently than tolbutamide and **does not interact** with salicylate or dicumarol.*
*Allopurinol **did not increase** the marrow toxicity of patients treated with cyclophosphamide, doxorubicin, bleomycin, procarbazine and/or mechlorethamine.*

	Total	**No DDI**	**with DDI**
Files:	579	164	415
Sentences:	5,806	3,762	2,044
Total of annotated DDIs:		3160	

Table 3.9: Basic Statistics on *annotated dataset* of the DrugDDI corpus.

	Avg. per document
Number of sentences:	10.03
Number of sentence that contain at least a DDI:	3.53
Number of sentences that contain no DDI:	6.50
Number of DDIs:	5.46 (0.54 per sentence)

Table 3.10: Average number of sentence and DDIs per document.

3.2.3.1. DDIAnnotate tool

Although, there exist several annotation tools for creating annotated biomedical corpora with such as BioNotate [Cano et al., 2009] or @Note [Lourenço et al., 2009], they do not allow to annotate DDIs in an adequate way. Thus, we decided to developed a client application DDIAnnotate (see figure 3.9) implemented in Java language, in order to help annotators in their task. The tool has an intuitive user interface where the documents are displayed and drugs drugs occurring in text are highlighted. There are six different colors for the highlighting, indicating each one of the semantic types that represent drugs. To annotate the interactions, the annotator only has to select a sentence, and then, to indicate the interacting drugs (comboxs *Drug 1* and *Drug 2*, and add -*green arrow*- the interaction to the list of interactions for this sentence. The annotator must add an interaction for each pair of interacting drugs. The user may also annotate the severity or the certainty degree for the drug interaction, however, we have not annotated this kind of information yet. We are planning to implement a new version to annotate

other information about drug-drug interactons such as the roles of each drug, doses, time course, among other features. We are also planning to study the BioNotate tool that allows to annotate documents in a collaborative way.
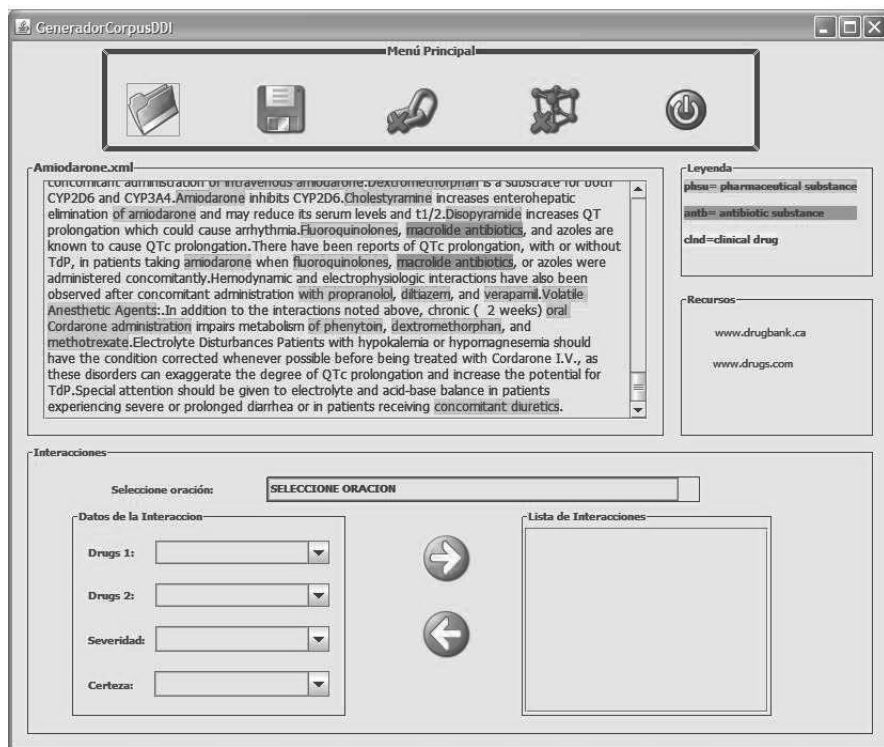
Figure 3.9: DDIAnnotate tool.

3.3. Conclusions

In this chapter, we have described the main corpora available in the biomedical domain showing that most biomedical annotated corpora for relation extraction are focused on the biological domain and DDIs are not covered. The size of these corpora is quite small and usually does not exceed one thousand sentences. The biomedical entities are automatically annotated, and then manually reviewed, but information about the annotation process is rather scarce. Most of the reviewed corpora do not have (or provide) any information about the inter-annotator agreement nor annotation guidelines. Regarding linguistic information contained within the corpora, LLL and BioInfer are the only corpora that contain syntactic information.

We have built the first annotated corpus for drug interaction. To the best of our knowledge, the problem of producing an annotated corpus for DDI extraction has not been explored to the depth and extent reported in this chapter, and the resulting corpus is the most richly semantically annotated resource for pharmacological text processing built to date. The DrugDDI corpus is available for research but cannot be used for commercial purposes. The availibility of this annotated corpus is a clear incentive for the development of drug interaction extraction approaches since DrugDDI provides a gold-standard data for their evaluation. A common shared corpus should increase the research and rapid advances in the field. The corpus consits of 579 documents from the DrugBank database. The average number of sentence per document is 10.3, and the average number of tokens per document is 211.5. DrugDDI contains a total of 3,160 DDIs that have been annotated at sentence level. The average number of interactions per document

is 5.46 and per sentence 0.54. DrugDDI corpus has been annotated with linguistic information including sentence boundaries, tokenization, phrase boundaries and phrase semantic classification provided by MMTx. Unfortunately, MMTx introduces several errors in its different levels of analysis, that we have not reviewed yet. Thus, some interactions have not been annotated because their drugs were not recognized by MMTx. We hope to encourage many researchers to make use of DrugDDI corpus for their research, and expect feedback from them that would be the most valuable source for further improvement of the corpus. Some of our future taks are:

1. Annotate the interactions at document level. Then, we will be able to evaluate the contribution of our anaphora resolution methods.

2. Review the linguistic information provided by MMTx.

3. Analyze the text with a parser trained on biomedical texts, such as Genia dependency parser [Sagae, 2007] or the parser proposed by Lease and Charniak [2005].

4. Increase the pharmacological information about the interactions including information such as the roles of the drugs, the mechanism of the action, the drug dosages, the time, the severety, among other.

5. Increase the size of the corpus.

6. Work with a major number of annotator and to measure the inter-agreement annotator as well as a comprehensive annotation guidelines that to be useful to other annotators.

7. Manually review the semantic annotation provided by MMTx with the assistant of several pharmaceutical experts as well as to measure the inter-annotator agreement.

8. Annotate negations.

Finally, we would like to note the EMEA corpus[3] as a possible source of information textual for the construction of annotated corpora for pharmaceutical domain. This is a parallel corpus that contains a total of 48,087 documents for 22 languages from the European Medicines Agency[4]. This corpus belongs to the collection of corpus provided by the OPUS project[5], which is an attempt to collect translated texts from the web, to convert and align the entire collection, to add linguistic annotation, and to provide the community with a publicly available parallel corpus. To our knowledge, the corpus EMEA does not contain any semantic information yet.

An interesting and challenging project is the Linking Open Drug Data (LODD)[6]. This project aims to link the various sources of drug data together in order the researchers, physicians and patients can take advantage of the connected data sets. In addition, we believe that the integration of data from these drug sources can be a valuable instrument in improving the performance of the biomedical NLP tasks, in particular, named entity recognition.

---

[3]http://www.let.rug.nl/ tiedeman/OPUS/EMEA.php
[4]http://www.emea.europa.eu/
[5]http://www.let.rug.nl/ tiedeman/OPUS/
[6]http://esw.w3.org/topic/HCLSIG/LODD

Chapter 4

Drug Name Recognition and Classification

4.1. Introduction

The recognition of drug names is not only an essential prerequisite step for the automatic discovery of DDIs from biomedical texts, but also required in other kinds of applications, such as Information Retrieval, Information Extraction, Information Management and new knowledge discovery in the pharmacological domain.

Drug name recognition aims to find drugs in biomedical texts and classify them into predefined categories (drug families), such as *analgesics*, *antihistamines*, *antivirals*, etc. This process is a challenging task, given the difficulties implied in biomedical text processing:

1. With the rapidly changing vocabulary, new drugs are introduced very frequently, while old ones are made obsolete. It is difficult to maintain and update terminological resources constantly. Although frequently updated, such resources cannot keep up with the accelerated pace of the changing terminology. Thus, systems capable of automatically detecting candidate terms for augmenting these resources would help in speeding up the time-consuming task of maintenance.

2. Naming conventions are available for a variety of domains in the biomedicine field, for example, in the pharmacological domain, the WHO International Nonproprietary Names (INNs) Program [WHO, 2006] defines a set of nomenclature rules to identify and classify generic drugs. However, these conventions are not strictly followed. Despite this fact, integrating this type of information can help in gaining basic insights into the underlying meanings of the drugs in question and, therefore, help in the classification of them.

3. Drug name recognition also requires a considerable linguistic analysis of drug entities. For example, an automatic IE system for DDIs should detects not only drug names occurring in the text, but also anaphoric expressions (such as *it* and *the drug*) that refer to interacting drugs. This problem will be tackled in chapter 5.

Figure 4.1 shows the pipeline architecture of our drug-drug interaction prototype. First, texts are processed by the MMTx program, which allows to recognize a variety of biomedical entities, among them drugs. Then, drugs found in such texts are classified into drug families . Over this basis, anaphora resolution is carried out to account for both nominal phrases referring to drugs and pronouns. Finally, the relation extraction module exploits the output of these previous modules in order to account for drug-drug interactions in biomedical documents.

This chapter describes the module for drug name recognition and classification, called *DrugNer*. This module extends the approach presented by Aronson [2001], in which MMTx is applied to recognize biomedical concepts in texts. Our module also exploits the shallow syntactic analysis of texts provided by MMTx. Subsequently, a set of nomenclature rules recommended by the WHO International Nonproprietary Names (INNs) [WHO, 2006] provides valuable insights to identify and classify drug names. First of all, the rules help in the identification of drug names that have not been detected by MMTx. In addition, these nomenclature rules can also classify drug names according to pharmacological or chemical

Drug	Date of Approval
Xiaflex (collagenase clostridium histolyticum)	February 3, 2010
Oleptro (trazodone)	February 2, 2010
Victoza (liraglutide) Injection	January 25, 2010
Ampyra (dalfampridine)	January 22, 2010
Actemra (tocilizumab)	January 8, 2010
Zyprexa Relprevv (olanzapine)	December 11, 2009
Kalbitor (ecallantide)	December 1, 2009
Agriflu (influenza virus vaccine, inactivated)	November 27,2009
Qutenza (capsaicin)h	November 16, 2009
Lysteda (tranexamic acid)	November 13, 2009
Istodax (romidepsin)	November 5, 2009
Pennsaid (diclofenac sodium)	November 4, 2009

Table 4.1: Recent drug approvals: brand names are shown first, followed by the generic names enclosed between parenthesis
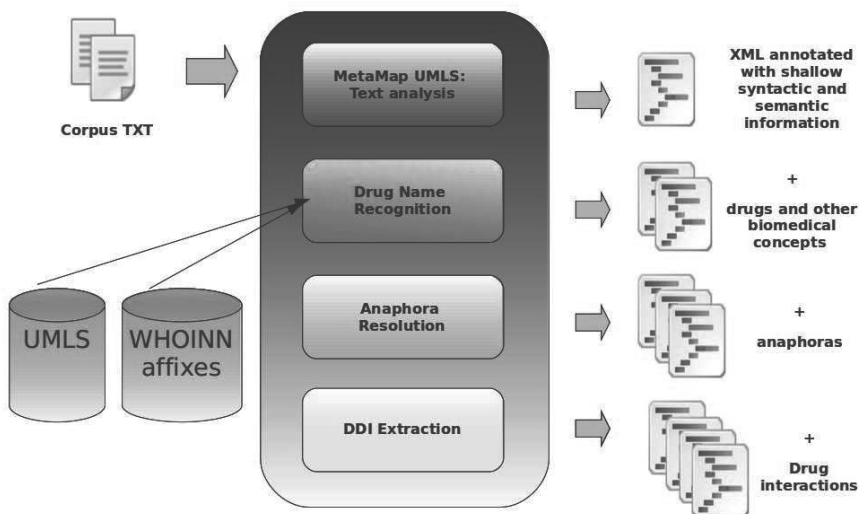
Figure 4.1: DDI Extraction prototype.

groups, that is, drug families. Drug families can represent a valuable clue for the detection of DDIs in biomedical texts. In the vast majority of cases, drugs that belong to the same family usually share the basic chemical structure and mechanism of action [Gilman et al., 1992, Russell, 2007], though there are exceptions. Therefore, if an interaction of a particular drug is known, there is a reasonable probability that another drug with similar chemical structure and metabolic pathway will also exhibit this interaction [Bottorff, 2006].

The chapter is organized as follows. The following section presents a brief summary review on related work in Biomedical Named Entity Recognition. The module for drug name recognition proposed in this thesis is described in section 4.3. The process of construction of the corpus used for the development and evaluation of DrugNer and experimental results are presented in section 4.4. We have developed a tool to highlight the pharmacological substances in texts, which is described in section 4.5. Section 4.6 offers some conclusions as well as unresolved issues.

4.2. Biomedical Named Entity Recognition

Biomedical Named Entity Recognition (BNER) is defined as the task of recognizing and categorizing entity names in biomedical domains. This task is a prior and essential step for mining useful knowledge from the biomedical literature. Identifying these biomedical entities is crucial for facilitating the retrieval of relevant documents and the identification of relationships between them, such as PPIs, DDIs, food-drug interactions, disease-gene relations, drug targets, among others.

Despite the availability of many well-known nomenclatures for biomedical entities, it is certainly a nontrivial task since these resources do not address certain issues such as changing vocabulary, ambiguity, synonymy and abundant use of acronyms, among others.

In this section, the current progress in biomedical name entity recognition is explained through some selected papers that clearly show methodological improvements. The papers are classified into the following three groups:

1. Dictionary-based approaches which try to find names of the terminological resources in the literature.

2. Pattern-based approaches which manually or automatically construct patterns to directly match them to candidate named entities in the literature.

3. Machine learning approaches which employ machine learning techniques, such as Hidden Markov Models (HMMs)

4. , Support Vector Machines (SVM) and Conditional Random Fields (CRFs) to develop statistical models for biomedical named entity recognition.

These approaches are, of course, not necessarily exclusive, indeed, some of the introduced papers merge two or more of them in order to deal with different aspects of the problem.

4.2.1. Dictionary-Based Approaches

Publicly available biomedical sources and databases such as UMLS [Humphreys et al., 1998], Gene Ontology [Ashburner et al., 2000], GenBank [Benson et al., 2007], FlyBase [Drysdale and Crosby, 2005], UniProt [Bairoch et al., 2005], Disease Database, among others, provide comprehensive lists of biomedical entities. This fact has promoted that most of the early approaches have been heavily dependent on such lists. In general, the basic idea of dictionary-based approaches is to match dictionary entries exactly against text. Thus, these approaches can provide ID information on recognized terms, which is very useful to integrate extracted information with data from information sources.

Krauthammer et al. [2000] propose a system for recognizing gene and protein names that used the BLAST tool [Altschul et al., 1997] for DNA and protein sequence comparison. First, the system translates both dictionary entries and input texts into sequences of nucleotides, which can be compared by BLAST, and then performs approximate string matching. Thus, it could recognize protein and gene names that have not been registered yet. The system was manually evaluated on a set of two papers, achieving an f-measure of 75%.

Hanisch et al. [2002] build a unified dictionary of genes and proteins by merging HUGO Nomenclature [Povey et al., 2001], OMIM database [Maglott et al., 2002], and UniProt. Texts are tokenized, in order, to look for each token in the dictionary. The method was evaluated on 470 Medline abstracts semi-automatically annotated, reporting an f-measure of 92.4%.

Tsuruoka and Tsujii [2004a] have focused on the problem of spelling variations. Biomedical named entities usually have many spelling variations, for example, the protein name *Anti-p24 (HIV-1) mono-clonal antibody CB4-1* can be represented with different spelling variants such as *monoclonal anti-p24 (HIV-1) antibody CB4-1*, *anti-p 24(HIV-1) monoclonal antibody CB 4-1* or *anti-p24(HIV1) monoclonal antibody CB41*. Two methods are proposed and evaluated on the Genia corpus [Ohta et al., 2002, Kim et al., 2003]. The first method uses approximate string searching techniques and achieves an f-measure of 64.7%. The second expands a protein name dictionary by a probabilistic variant generator and achieves a f-measure of 66.6%.

Sirohi and Peissig [2004] compare the use of various drug lexicons to automatically extract medication information from electronic medical records. Three drug lexicons are used as sources for medication extraction: first, containing drug name and generic name; second with drug, generic and short names; third with drug, generic and short names followed by filtering techniques. The effect of each lexicon is evaluated on a collection of 100 documents which contains a total of 641 medications. Extraction with

the first drug lexicon resulted in 85.2% recall and 96.9% specificity (this metric is defined in section 2.2). The integration of the short names increases the recall to 96.4%, but decreases the specificity to 80,1%. Finally, the use of a set of filtering techniques with the previous lexicons increases the specificity to 98.8% while slightly decreasing the recall to 95.8%. Thus, this combination extracts the highest number of medications while keeping the number of extracted non-medications low.

The method proposed by Schuemie et al. [2007] combines information from several gene and protein databases such as Entrez Gene [Maglott et al., 2006], the Mouse Genome Database (MGD) [Blake et al., 2003], FlyBase and *Sacchromyces* Genome Database [Cherry et al., 1998] for automatic generation of a comprehensive dictionary. Schuemie et al. [2007] also use a set of rules to generate spelling variations. Convert numbers to greek letters, remove the word delimiter between letters and numbers or rewrite numbers as roman numerals are some examples of the rules proposed. The BioCreAtIvE corpus [Hirschman et al., 2005] and the Gena dataset [Koike and Takagi, 2004] are used to evaluate the method. The results show that the rules proposed achieve higher levels of recall without sacrificing precision. In addition, high recall levels are achieved by the combination of different databases, and applying these spelling-variation rules.

Yang et al. [2008] propose a dictionary-based approach that expands a bioentity name dictionary by the identification of abbreviations. Some post-processing methods are also applied including Pre-keyword and Post-keyword expansion, Part of Speech (POS) expansion, merge of adjacent bio-entity names and the exploitation of contextual clues. The experiments are conducted using the JNLPBA2004 [Kim et al., 2004], achieving a f-measure of 68.80%.

More recently, a dictionary for the identification of small molecules and drugs has been developed in [Hettne et al., 2009]. The dictionary combines information from UMLS, MeSH [Elwood et al., 1948, Lipscomb, 2000], ChEBI [Degtyarenko et al., 2008], DrugBank [Wishart et al., 2006, 2007a], KEGG [Kanehisa and Goto, 2000], HMDB [Wishart et al., 2007b], and ChemIDplus[Tomasulo, 2002]. The combined dictionary achieves a precision of 67%, recall 40% (recall is 80% for trivial names).

Many existing systems adopt a hybrid approach for improving the performance of BNER systems by combining methods from two or more approaches. Tsuruoka and Tsujii [2004b] developed a two-phase method to address the limitations of the dictionary-based approaches. Firstly, a probabilistic generator produces morphological variations of protein names included in the UMLS Metathesuarus. Then, a protein name dictionary is expanded by scanning texts for protein name candidates. Finally, a Naïve Bayes classifier is used to filter out the irrelevant candidates of short names. The method is evaluated on the Genia corpus and achieves an f-measure of 66.6% with a precision of 71.7% and a recall of 62.3%.

Abbreviations and acronyms are frequently used in the biomedical texts to rename drugs and other concepts. The high ambiguity of these terms and the lack of acronym dictionaries turn the automatic resolution of them into a very hard task. Gaudan et al. [2005] combine an automatic analysis of Medline abstracts and linguistic methods to build a dictionary of abbreviation/definition pairs. The dictionary is used for the resolution of abbreviations occurring with their long-forms. Ambiguous global abbreviations are resolved using SVM that have been trained on the context of each instance of the abbreviation/sense pairs, previously extracted for the dictionary. The system achieves a precision of 98.9% and a recall of 98.2%.

Recently, an innovative service, REFLECT [Pafilis et al., 2009], has been developed by European Molecular Biology Laboratory (EMBL) to improve access to biological information in context. This service uses a dictionary to recognize gene, protein and small molecule names. Each recognized entity is linked to documents that show additional information such as synonyms, database identifiers, sequence, domains, 3D structure, interaction partners, subcellular location, and related literature. In addition, it lets end-users systematically tag gene, protein, and small molecule names in any web page. The service can be installed as a plug-in in the navigators.

Dictionary-based methods are very precise, however, they present several drawbacks such as low recall due to synonyms and spelling variations, and the inability to discover newly published names that have not yet covered by any resource. In addition, false recognition mainly caused by short and ambiguous names degrades overall precision.

4.2.2. Pattern-Based Approaches

While dictionary-based approaches can deal only morphological variations, pattern-based approaches deal with a broader range of variations such as word-order and syntactic variations.

Basically, pattern-based approaches exploit the contexts in which the entities appear in the text. These contexts can be discovered and expressed as patterns in order to locate new entities.

Fukuda et al. [1998] develop the PROPER (PROtein Proper-noun phrase Extracting Rules) system for protein name recognition based on the use of simple lexical patterns and orthographic features. The method first identifies *core terms*, those which contain special characters (such as upper cases, commas, hyphens, slashes, brackets or digits), and *feature terms*, those which describe biomedical functions of compound words (for example, *protein* and *receptor*). Core and feature terms are then concatenated by utilizing hand-crafted patterns and the boundaries are extended to adjacent nouns and adjectives. The method is evaluated on 30 abstracts achieving 94.70% precision and 98.84% recall (f-measure=96.7%).

The AbGene system [Tanabe and Wilbur, 2002] is one of the most popular pattern-based approaches for BNER, in which the Brill tagger [Brill, 1992] is adapted to recognize single word protein and gene names. A set of 7000 sentences is manually annotated with gene and protein entities. This corpus (called ABGene corpus) is used to define a set of hand-built patterns based on lexical features to filter out false positives and recover false negatives. The patterns are applied to identify the context in which protein and gene names are used. The system achieves a precision of 85.7% and a recall of 66.7%.

The YAPEX system, presented in [Franzén et al., 2002], combines lexical and syntactic information, heuristic rules and a document-local dynamic dictionary. The syntactic information is used to identify single and multi-word protein names. A corpus of 200 abstracts, called as YAPEX, is manually annotated to evaluate the system. Its test corpus consists of 101 MedLine abstracts, containing a total of 1936 annotated protein names. The system achieves an f-measure of 67.1% (recall of 61.0% and a precision of 62.0%).

Nenadic et al. introduce an integrated framework for terminology-driven mining from biomedical literature. The framework integrates the following components: automatic term recognition, term variation handling, acronym acquisition, automatic discovery of term similarities and term clustering. The term variant recognition takes into account orthographical, morphological, syntactic, lexico-semantic and pragmatic term variations. In particular, they address acronyms as a common way of introducing term variants in biomedical papers. Term clustering is based on the automatic discovery of term similarities. They use a hybrid similarity measure, where terms are compared by using both internal and external evidence. The measure combines lexical, syntactical and contextual similarity. Experiments on terminology recognition and clustering have been performed on a corpus of 2082 Medline abstracts related to nuclear receptors. They achieve around 99% precision at 74% recall in acronyms recognition, and around 70% precision in clustering semantically similar terms.

Seki and Mostafa [2005] propose a hybrid method that combines a set of hand-built patterns, a dictionary and a probabilistic model for locating complete protein names. Words such as *binding*, *related*, *associated*, as well as suffixes such as *-ine*, *-tide* or *-yl* are some of the surface clues that are used to define the patterns. The probabilistic model uses word classes based on suffixes and word structure for dealing with data sparseness problem. The dictionary is compiled from the Swiss-Prot and TrEMBL protein databases [Boeckmann et al., 2003]. The method is evaluated on the YAPEX corpus, achieving an f-measure of 63.3% (recall of 66.9% and precision of 60.1%).

In the approach presented in [Torii et al., 2004], a set of patterns based on features that appear within the entity name as well as contextual information is manually defined to classify the entities. The experiments were conducted on the Genia Corpus. The approach achieves an f-measure of 86% with a precision of 87% and a recall of 86%. Later, Torii and Liu [2006] evaluated the performance of headwords and suffixes in predicting semantic classes of biomedical terms. New semantic classes are defined by modifying an existing UMLS semantic group system and incorporating the Genia ontology. Headwords and suffixes that are significantly associated with a specific semantic class are defined. The terms in UMLS are reclassified by these semantic headwords and suffixes. The performance of semantic assignment using semantic suffixes reached an f-measure of 86.4% with a precision of 91.6% and a recall of 81.7%.

The PASTA (Protein Active Site Template Acquisition) [Humphreys et al., 2000, Gaizauskas et al., 2003] is a pipeline of processing components that performs the following major tasks: tokenization, sentence splitting, lexical lookup of terms in terminological lexicons to identify and correctly classify instances of the term classes, syntactic and semantic processing of terms, construction of a discourse model of the input text based on the semantic representation of each sentence, and finally, a template extraction module that looks in the discourse model and outputs the templates. The PASTA system achieved an f-measure of 85% on a set of 61 abstracts.

The system presented in Caporaso et al. [2007] uses regular expressions to identify mutations from

text. They built a development data set (305 abstracts containing 605 mutations), and a test data (with 910 mutation mentions from 508 abstracts). Annotation is performed with Knowtator [Ogren, 2006], using an ontology that was developed for describing point mutations. The development data set is used to design the regular expressions. The systems achieves a precision of 98.4%, a recall of 81.9%, and an f-measure of 89.4%.

Pattern-based approaches have achieved remarkable performance. However, the main drawback of these approaches is the need to manually construct patterns that must be defined by domain experts. This is a time-consuming task. Xu et al. [2008] propose a bootstrapping system for learning pattern that allows to build a disease dictionary from abstracts. The system extracts diseases by matching the set of seed patterns in the parse tree of the sentences. Then, it also discovers new patterns from the extracted diseases and adds them to the set of seed patterns. The system repeats this process a fixed number of interactions. The built dictionary consists of 1,922,283 disease names. The dictionary is used to identify disease names in 100 manually annotated abstract. It achieves a precision of 80%, a recall of 78% and an f-measure of 81%.

Another of the shortcomings of the pattern-based approaches is that they are not flexible and cannot easily adapt to identify other biomedical entities, because naming conventions are often very different in the biomedical subdomains.

4.2.3. Machine Learning Approaches

In [Collier et al., 2000], a Hidden Markov model (HMM) is trained with bigrams based on lexical and character features. Experiments are performed on a corpus of 100 abstracts manually annotated by domain-experts with term classes such as proteins and DNA. The model achieves an F-measure of 73%.

The approach presented in [Lee et al., 2003] is based on the use of SVM. It was evaluated on the Genia corpus, and reported an f-measure of 69.2%. NLProt [Mika and Rost, 2004] is a hybrid system that combines a pre-processing dictionary and rule-based filtering step with several separately trained SVM models to identify protein names. The system achieves an f-measure of 75% on the YAPEX corpus.

The system PowerBioNe [Zhou et al., 2004] uses a HMM based named entity recognition that integrates various features such as word formation pattern, morphological pattern (prefix and suffix) and part of speech tags. The system also uses a k-NN algorithm to resolve the data sparseness problem. In addition, the system provides a pattern-based method to automatically extract rules to deal with the cascaded entity name phenomenon. The system is evaluated on the Genia corpus and achieves a f-measure of 66.6%.

Kou et al. [2005] propose two hybrid methods, SemiCRFs and dictionary HMMs. SemiCRFs is an extension to Conditional Random Fields (CRF) that enables a more effective use of dictionary information as features. Dictionary HMMs convert a dictionary to a large HMM that recognizes phrases from the dictionary, as well as variations of these phrases. These methods are evaluated on the YAPEX corpus achieving an f-measure of 66.1% and 51% respectively.

Takeuchi and Collier [2005] apply SVM for the identification and semantic annotation of scientific terminology in the domain of molecular biology. The model uses various features such as surface words, orthographic features, head noun features and contextual information. The model achieves an f-measure of 74% on a set of 100 abstracts.

Bunescu et al. [2005] compare various methods for the recognition of protein names such as SVMs, Maximum Entropy, Memory-based Learning (MBL) or Trans
formation-based Learning (TBL), RAPIER (a system for relational learning of pattern match rules) [Califf and Mooney, 1999], among others. These methods are evaluated on 748 abstracts which are manually annotated with human genes and proteins. The best performance is achieved by the Maximum Entropy method with an f-measure of 57.9%.

Dimililer and Varoglu [2006] apply SVM to the identification and automatic annotation of biomedical named entities in the domain of molecular biology. They use and compare well-known features such as word formation patterns, lexical, morphological, and surface words on recognition performance. The method is evaluated on the JNLPBA 2004 corpus, achieving an f-measure of 69.87%.

Wu et al. [2006] also present a recognition system based on SVM. The features set is a combination of lexical and contextual features. A corpus of 100 abstracts in the domain of molecular biology is collected from Medline and annotated with gene names. The experiments show that the system achieves an f-measure of 81.57% with a precision 81.40% and a recall of 81.74%.

Ponomareva et al. [2007] have developed a HMM-based biomedical NER system that takes into account only parts-of-speech as an additional feature. This morphological information allows to detect the entity

boundaries and to tackle the problem of nonuniform distribution among biomedical entity classes. The system achieves an f-measure of 65.7% with a precision of 62.4% and a recall of 69.4%.

Vlachos [2007] compares two such systems, one based on a HMM and one based on CRF and syntactic parsing. Three different corpora from the FlyBase database have been built. The first corpus consits of 5 full papers and the second one of 16,609 abstracts. Both corpora are parsed by the RASP parser [Briscoe et al., 2006] and automatically annotated with gene names using longest-extent pattern matching. The third corpus (described in [Vlachos and Gasperin, 2006]) consists of 82 abstracts that are annotated by a computational linguist and a FlyBase curator . The method based on a HMM achieves an f-measure of 81.86% with a precision of 89.14% and a recall of 75.68% and overcomes the hybrid method combining CRF and syntactic parsing, which obtains a f-measure of 74.72%, a precision of 90.89% and a recall of 63.43%.

BioCreative I [Yeh et al., 2005] and II [Smith et al., 2008] gene mention tasks have been of crucial importance to encourage research on gene name recognition. A corpus of 20,000 abstracts is selected and annotated for training and testing purpose with the AbGene tagger [Tanabe and Wilbur, 2002]. The best systems were based on the use of machine learning techniques. The best performance in BioCreative II is obtained by the system [Ando, 2007], achieving an f-measure of 87.2% (the highest achieved f-measure for the previous task was 83.6%). The system used a general purpose named entity recognition framework [Ando and Zhang, 2005]. This recognizer consists of a regularized linear classifier trained utilizing standard features such as word strings and character types. Other participating systems [Kuo et al., 2007, Huang et al., 2007, Klinger et al., 2007] were based on CRF, or in a combination several machine learning methods such as CRFs and SVM. A detailed description of the participating systems can be found in [Smith et al., 2008] and [Yeh et al., 2005].

The system Banner [Leaman and Gonzalez, 2008] is an open-source system based on CRF method and the use of orthographical, morphological and shallow syntax features. The system is evaluated on the corpus created for the BioCreative 2 Gene Mention Tagging Task [Wilbur et al., 2007, Smith et al., 2008], achieving a precision of 85.09% and a recall of 79.06%. One of its major advantages is that it does not make use of neither semantic features nor domain rules, so maximizing domain independence.

Recently, Gurulingappa et al. [2009] have developed a system that combines IE and machine learning techniques to classify drugs. Terms expressing information about drugs are used as features within a machine learning framework to predict class labels from the anatomical therapeutic chemical (ATC) classification for unclassified drugs. The system is tested on a portion of the ATC classification containing drugs, achieving an accuracy of 77.12%.

Machine learning techniques are able to identify biomedical entities without human intervention. However, in order to achieve good recall, these techniques need large annotated corpora. Thus, they depend heavily on the annotated corpora for training and testing. Corpus annotation is expensive work, usually involving the need of domain experts, extensive time and labor. Few corpora are available for biomedical named entity recognition. They are usually limited to protein and gene, usually exhibit a small size and do not contain annotations on other types of biomedical entities, such as drugs. Therefore, these limitations have made it difficult to apply supervised machine learning techniques to recognize biomedical entities.

4.2.4. Unsolved Issues in Drug Name Recognition

The analysis of the state of art shows that while most studies of BNER have mainly focused on genes and proteins, the drug names have not widely addressed. There are several open issues such as ambiguous names, synonyms, variations and newly published names related to the drug name recognition. Some challenges of this task are:

1. Face anaphora problem: drug names as well as other biomedical named entities can be expressed in various linguistic forms including plurals, compounds and anaphoric expressions. Therefore, drug name recognition also requires a considerable linguistic analysis. For example, an automatic information extraction system for DDIs should detect not only drug names occurring in the text, but also anaphoric expressions, such as *it* and *the drug*, that refer to interacting drugs, as illustrated in example 6. Thus, the system must handle the fact that a possible interaction may occur between *levofloxacin* and *warfarin*, by resolving the noun phrase *this drug* with the drug name *levofloxacin*. Such anaphoric expressions subsequently may be replaced with their antecedent drug names as defined in the preceding context. This problem will be tackled in the following chapter 5.

2. Face management of terminological resources: with the rapidly changing vocabulary, new drugs are

continually created while older ones are made renamed, which makes difficult to keep up to date the terminological resources.

3. Face synonymy problem: drug names are synonymous with other drug names. For example, *adofen, affectine, alzac, ansilan, deproxin, erocap, fluctin, fluctine, fludac, flufran, flunil*, and 27 other trade names all are brands of the generic drug *fluoxetine*.

4. Face ambiguity problem: Drug names often have the same name as an common English word such as *Because* (a contraceptive) or *Duration* (nasal spray). Thus, terms in free text may be ambiguous and resolve to multiple senses, depending on the context in which they are used.

5. Face acronyms and abbreviations problem: abbreviations and acronyms are frequently used in the biomedical texts to rename drugs and other concepts. The high ambiguity of these terms and the lack of acronym dictionaries turn the automatic resolution of them into a very hard task.

6. Interpret drug naming conventions: drug naming conventions are available, however, these conventions are not strictly followed. Despite this fact, integrating this type of information can help gaining basic insights into the underlying meanings of the terms in concern, and, therefore, helping in the classification of the terms.

Regarding the drug name classification, the drugs can be categorized in different ways according to their mechanism of action, their indications or their chemical structure. Currently, there are several drug classification systems such as the Anatomical Therapeutic Chemical (ATC) Classification System[1] defined by the World Health Organization (WHO) or the AHFS Pharmacologic-Therapeutic Classification[2] used in hospitals in the United States. Each classification system has its advantages and limitations and its usefulness depends on the purpose, the setting used and the user's knowledge of the methodology. Among the various systems proposed over the years, only the ATC system [WHO, 2003] has survived to attain a dominant position in drug utilization research worldwide. ATC divides drugs into different groups according to the organ or system on which they act and/or their therapeutic and chemical characteristics. However, ATC does not provide comprehensive coverage of all generic drugs and the brand drugs are not classified in it. On the other hand, electronic biomedical resources such as UMLS yield semantic categorizations too broad to classify concepts of specific domains such as the pharmacological ones.

The pharmacological and/or chemical group to which a drug belongs can be an essential clue to automatically detect information regarding its interactions or adverse effects. In the vast majority of cases, drugs that belong to the same family usually share the basic chemical structure and action mechanism [Russell, 2007], although there are exceptions. Thus, if the interaction of a particular drug is known, it is quite likely that another drug with similar chemical structure and the same pathway of metabolism will show the same interaction [Bottorff, 2006]. Therefore, providing a comprehensive and suitable drug classification framework can be crucial to successful extraction of DDIs.

Example 6 Sentence containing an anaphoric expression (*this*) that refers to the drug name *levofloxacin*

Levofloxacin, a fluoroquinolone, is one of the most commonly prescribed antibiotics in clinical practice. Several case reports have indicated that *this drug* may significantly potentiate the anticoagulation effect of warfarin.

4.3. DrugNer: drug name recognition and classification

This section describes the technique followed for drug name recognition and classification, called DrugNer. Based on the approach [Aronson, 2001], we decided to study the performance of the MMTx program to identify drug names in texts. We needed an additional tool or method to classify the drugs into drug families, because MMTx (more specifically the UMLS semantic network) provides a too broad classification, only distinguishing between antibiotics and the rest of pharmacological substances.

Basically, DrugNer combines the information obtained by the MMTx program and nomenclature rules recommended by the WHO International Nonproprietary Names (WHOINN) Program [WHO, 2006] to identify and classify drug names. As it is was explained in the previous chapter, MMTx allows to analyze the text syntactically splitting it into components including sentences, paragraphs, phrases,

---

[1]http://www.who.int/classifications/atcddd/en/
[2]http://www.ahfsdruginformation.com/class/index.aspx

lexical elements and tokens. Moreover, MMTx tries to link the text of each phrase with some concept from the UMLS Metathesaurus. Concepts are classified with at least one of the semantic types from the UMLS semantic network. The definition of these semantic types as well as a more detailed description of the analysis provided by MMTx can be found in chapter 3. This way, MMTx allows to identify drugs and other biomedical concepts. In the experiments presented in this chapter, DrugNer only considers the semantic types *pharmacological substance (phsu)* and *antibiotic (antb)* to identify the terms that refer drugs. Once texts have been processed by MMTx and terms occurring in the text annotated and related to concepts of the UMLS Metathesaurus, a rule-based module classifies the drug names occurring in texts in pharmacological or chemical families.
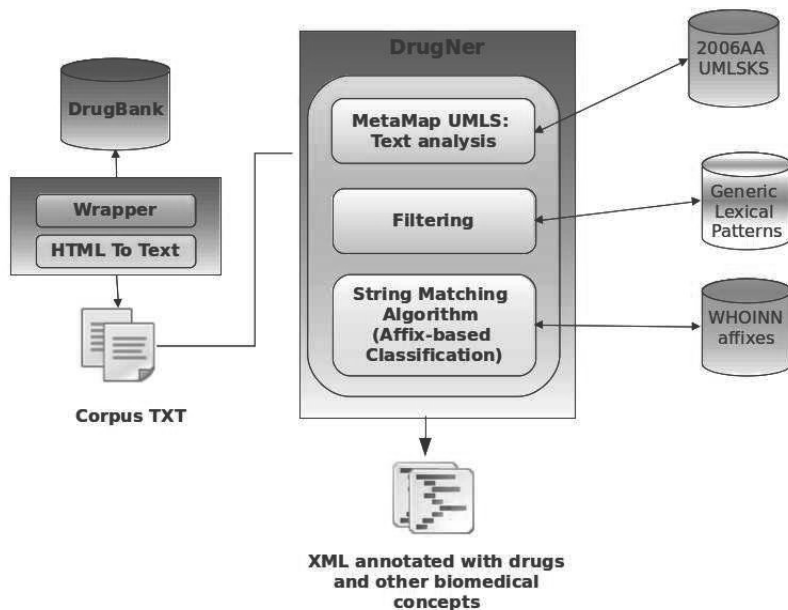
Figure 4.2: DrugNer architecture

The rules are based on the common affixes (also called as *stems*) selected and defined by the WHOINN program. These common affixes, currently in use, represent classes of substances that are pharmacologically or chemically related to. By using common affixes the medical practitioner, the pharmacist, or anyone dealing with pharmaceutical products can recognize that the substance belongs to a group of substances having similar pharmacological activity or chemical structure. Table 4.3 shows some of the affixes used in the classification of drug names. The full list can be obtained from the document [WHO, 2006] which contains the affix classification system used by the INN Program to categorize the main activity of pharmaceutical substances.

The affix classification system consists of two different types of categorizations: pharmacological or chemical. Thus, it is less consistent than the Anatomical Therapeutic Chemical (ATC) classification system in which the drugs are divided into different groups according to the organ or system on which they act and their chemical, pharmacological and therapeutic properties. Despite the inconsistencies of the WHOINN classification, we have decided to use it because it provides the affixes as well pharmacological and chemical information that could also be very useful in predicting DDIs occurring in texts. The term *drug family* is frequently used and can be interpreted as *pharmacological group*, though it is also often used to designate *chemical group*. We have decided to use the term *family* instead of *group*, since this term is broader and more general and because it allows us to include both the pharmacological and chemical groups.

The affixes together with their corresponding pharmacological or chemical groups are compiled in a list. This list is scanned in order to build the suitable regular expression for each affix. For example, for the affix *-flurane*, the regular expression should be *[A-Za-z0-9]*flurane*, so any alphanumeric string that ends with the suffix *-flurane* is recognized by this regular expression. Similarly, for the affix *-adol-*, the regular expression should be *[A-Za-z0-9]*adol[A-Za-z0-9]**.

Affixes	Regular Expression
-flurane	[A-Za-z0-9]*flurane
-bersat, -toin	[A-Za-z0-9]*[bersat|toin]
-giline, -moxin	[A-Za-z0-9]*[giline|moxin]

Table 4.2: Examples of regular expressions for drug name recognition.

Once the regular expressions have been built, the text of each phrase is matched against the regular expressions in order to detect the possible affixes that can classify the phrase. In the case in which several regular expressions can be matched with the input text, the longest affix is selected (see table 4.3). When a correct affix is found, appropriate information about the pharmacological or chemical family and the definition associated with the affix is added to the phrase. In addition, its corresponding information from UMLS Metathesaurus (CUI, definition, semantic types) is also extracted. This process could be configured to keep all the candidate affixes, and the selection of the most suitable affix could be made by the end user of the DrugNer Viewer tool (see section 4.5).

Although MMTx data is updated each year, the latest available release of MMTx (2.4.C) is on the basis of the 2006AA UMLS knowledge sources (March 2006). Thus, MMTx cannot detect those new concepts that have been included in UMLS after that date. Therefore, our main hypothesis is that the affixes recommended by the WHOINN not only allow the classification of the drugs, but also could help to find possible new drug candidates that have not been detected by MMTx. Thus, the affixes were applied on two different set of phrases. The first set is made up of phrases for which MMTx did not found any candidate concept in UMLS. These phrases may be possible new candidates for drug names, that are not included in UMLS Metathesaurus. The second set consists of phrases that have been classified by MMTx as pharmacological substances (*phsu*) or antibiotics (*antb*).

4.4. Evaluation

In this work, the Medline bibliographic database is used as the main data source because of its wide coverage of biomedical sciences and its public availability. A corpus of 849 abstracts is compiled from PubMed using the phrase *drug-drug interaction*.

MMTx detects a total of 106,576 phrases. 7.5% of them (8,093 phrases) are categorized as *pharmacological substances* (7,691) or as *antibiotics* (402) in UMLS. Of those phrases, 49.8% (53,037) belong to other semantic types such as *organic chemical*, *lipid*, *carbohydrate*, among others, which are out of the scope of the present study. For the rest of the phrases, (45,449), MMTx did not find any concept in UMLS that covered them. Table 4.4 shows the main characteristics of the compiled corpus for the evaluation of DrugNer.

In order to identify new candidates of generic drugs not detected by MMTx, the affix-based module is applied to the phrases that have not been detected by MMTx (45,449), identifying 255 initial candidates. A pharmacist manually evaluated this set, ruling out 74 phrases (false positives introduced by the affixes) and declaring the rest, 181, as generic drugs. Some of the identified drugs are presented in table 4.4.

To calculate the total coverage of our system it is necessary to take into account those drugs that have not been detected either by MMTx or by affixes (false negatives). Due to the excessive number of phrases (45,194) to evaluate, a set of lexical patterns was used to filter terms such as numeric expressions, verbs, adverbs and common nouns of biomedical domain, reducing the set to 5,964 phrases. Finally, a manual evaluation shows that only 20 of them are drugs. Table 4.4 shows some of them.

Precision and recall are standard measures for evaluating the performance of Information Retrieval and Extraction Systems. In our case, recall can be described as the ratio between a number of correctly recognized drugs and all the drugs occurring in the corpus. Precision is the ratio between the number of correctly recognized drugs and all the drugs recognized by DrugNer (see table 4.4). Table 4.4 shows the overall performance obtained using only MMTx and combining MMTx and the affix-based classification.

Affixes	Family
-flurane	General anaesthetics, volatile
-bersat, -toin	Anticonvulsants
-adol-, -azocine, -eridine, -ethidine, -fentanil, nal-	Narcotic analgesics
-ac, -adol-, -arit, -bufen, -butazone, -coxib, -icam, -fenamate, -nixin, -profen, -metacin, -adom, -fenine	Analgesics-antipyretics
-fylline, -racetam, -vin-	Analeptics
-azenil, -azepam, -bamate, -carnil, -peridone, -perone, -pidem, -plon, -pride, -quinil, -spirone, -zafone	Anxiolytic sedatives
-perone	Antipsychotics (neuroleptics)
-oxetine	Antidepressants
-giline, -moxin	MAO inhbitors
-pin(e), -pramine, -triptyline	Tricyclic antidepressants
-anserin, -setron	Serotonin receptor antagonists
-caine	Local anaesthetics
-curium, -ium	Neuromuscular blocking agents
-azoline, -drine, -frine, -terol	Adrenergic agents
-serpine	Adrenergic neurone blocking agents
-verine	Spasmolytics, general
-afil, -dil, -entan	Vasodilators
-dipine, -fradil, -pamil, -tiazem	Coronary vasodilators, also calcium channel blockers
-nicate	Peripheral vasodilators
-astine	Antihistaminics
-tadine, -tidine	Histamine H1, H2 receptor antagonists
-bradine, -denoson, -vaptan	Cardiovascular agents
-dan, -rinone, -afenone	Cardiac glycosides and drugs with similar action
-afenone, -aj-, -cain-, -ilide, -isomide, -kalant	Agents influencing heart muscle excitability and conductivity
-azosin, -dralazine, guan-, -kalim, -kiren, -(o)nidine, pril(at), -sartan	Antihypertensives
-fibrate, -nicate, -vastatin	Antihyperlipidaemic drugs
-cog, -cogin, -fiban, -gatran, -parin	Agents influencing blood coagulation
-arol, -grel-, -irudin, -pafant, -troban	Anticoagulants

Table 4.3: Some affixes recommended by WHOINN.

Drug	Suitable affixes	Most suitable affix
Azelnidipine	-dipine, -pine, -ine, -ni-	-dipine
Lopinavir	-navir, -vir-	-navir
Amiodarone	-arone, -one, -io-	-arone
Minocycline	-cycline, -ine	-cycline
Sulfinpyrazone	-azone, -zone, -one	-azone
Aripiprazole	-piprazole, -prazole	-piprazole
Furafylline	-fylline, -ine	-fylline
Gemcitabine	-citabine, -abine, -ine	-citabine
Mometasone	-metasone, -one	-metasone
Simvastatin	-vastatin, -stat-	-vastatin

Table 4.4: Examples of matching phrases and affixes.

An important contribution of this work is the classification achieved by the affixes recommended by the WHOINN Program that MMTx is not able to provide. Our hypothesis is on the basis of the idea that the affix-based classification could allow detection of the pharmacological or chemical family of the drugs classified as pharmacological or as antibiotics by MMTx, achieving, in this way, a more informative and suitable categorization of them. Initially, the affixes are able to classify 48.5% (3,926) of the phrases that are detected and categorized as drugs by MMTx (8,093). In order to assess the precision

Characteristics	Number
Abstracts	849
Sentences	10,146
Phrases	106,579
Phrases not classified by MMTx	45,449
Phrases classified as *phsu* or as *antb* in UMLS	8,093
Phrases classified with other semantic types in UMLS	53,037

Table 4.5: Characteristics of DrugNer corpus.

Name	Affix	Family	Num
Ciclofenac	-ac	Antiinflammatory	5
Efepristin	-pristin	Antibacterial	7
Armodafinil	-nil	Anxiolytic sedatives	10
Dabigatran	-gatran	Antithrombotic agents	3
Aplaviroc	-vir-	Antivirals	1
Maraviroc	-vir-	Antivirals	5
Vicriviroc	-vir-	Antivirals	3
Darunavir	-navir	Antivirals	39
Dasatinib	-tinib	Antineoplastic agents	7
Sunitinib	-tinib	Antineoplastic agents	28
Nilotinib	-tinib	Antineoplastic agents	2
Vorinostat	-inostat	Histone deacetylase inhibitors	7
Sitagliptin	-gli-	Oral antidiabetics	7
Tanespimycin	-mycin	Antibiotics	5

Table 4.6: Examples of drugs detected only by the affix-based classification.

of the affix-based classification, the pharmacist manually evaluates the phrases. The ATC classification system and other drug information resources are used to assist in this evaluation. In this case, precision is defined as the ratio between the number of correctly classified drugs and all the classified drugs by the affix-based module. The evaluation shows that 2,941 have been correctly classified by the affixes as opposed to 355 that have been wrongly classified. In other words, the affix-based classification obtains an precision rate of 74.9%. Short stems such as *-pin* (tricyclic antidepressants), *-ol* (alcohols and phenols), *-ox* (oral antiacids), *-ni-* (nicotinic acid or nicotinoyl alcohol derivatives) are responsible for the incorrect classifications. Thus, research in additional clues is necessary to detect these drug families.

4.5. DrugNer Viewer tool

A prototype, called DrugNer Viewer, has been developed; it is a visual tool that highlights the pharmacological substances in texts. DrugNer Viewer allows exploration of folders and selection of text files. Once the end user has selected a file, DrugNer processes it to detect and classify the drugs. Finally, DrugNer Viewer shows the content of file and highlights the drugs occurring in the text (figure 4.3). In

Name	Drug Family
Posaconazale	Triazole drug
Rapamcyin	Immune suppression drug
Gadobenate dimeglumine	Contrast agent for magnetic resonance
Riluzole	Nervous system drugs
2-Methoxyoestradiol	Angiogenesis inhibitors

Table 4.7: Examples of drugs detected neither by MMTx nor by affix-based classification.

Drugs detected by MMTx	8,093	(97.6%)
Drugs only detected by affixes	181	(2.2%)
Drugs detected neither by MMTx nor by stems	20	(0.2%)
Total:	8,294	

Table 4.8: Drugs in the corpus.

	Recall(%)	**Precision(%)**	**F-measure(%)**
MMTx	97.5	100	98.73
MMTx + affixes	99.8	99.1	99.45%

Table 4.9: Overall performance of the DrugNer module.

addition, the user can select any of the highlighted drugs, and then, DrugNer Viewer shows information concerning the selected drug (definition, affixes used in its classification and other information from UMLS). DrugNer Viewer can become a valuable tool for healthcare professionals and scientists, allowing them easy and rapid access to relevant information about drugs.
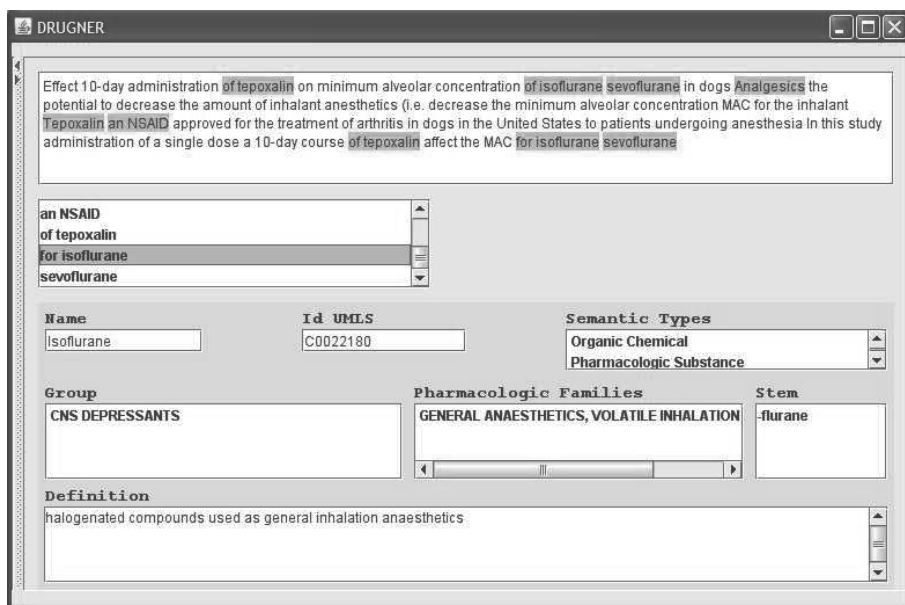
Figure 4.3: DrugNer Viewer tool

4.6. Conclusions

Detecting and classifying drug names occurring in biomedical text is a valuable task in drug discovery knowledge. It is a challenging task due to the difficulties implied in biomedical text processing such as such as ambiguous names, synonyms, variations and newly published names. In this chapter, we have presented a hybrid aproach that combines the use of the MMTx tool and a set of nomenclature rules to identify and classify drug names. MMTx is an effective program for the automatic processing of the biomedical texts and has been used extensively for text mining applications in the biomedical domain [Aronson, 2001, Li and Wu, 2006, Reeve et al., 2007]. However, MMTx is not able to provide

complete and useful information about pharmacological substances. The use of affixes recommended by the WHOINN Program helps to detect drugs and establish suitable information about the drugs such as the pharmacological or chemical family. Although evaluation reveals that using affixes in isolation is not feasible enough in detecting drugs they help to improve the coverage. The affix-based module identifies 255 initial candidates of whom 74 are not pharmacological substances. Most of these wrong identifications are given by the short stems that are too ambiguous to correctly detect drugs. In our experiment, we assumed that the drug name recognition provided by MMTx was correct, because our main objective was not to evaluate the performance of MMTx, but to study if the affixes could help to identify new drugs not detected by MMTx. In future work, we are planning to provide a more realistic evaluation taking into account the mistakes made by MMTx. It is probable that the performance will be affected by these mistakes.

As outlined previously, the affixes are able to correctly classify 74.9% of drugs occurring in the texts. The list of used affixes is not exhaustive and does not cover all pharmacological or chemical families. Each year, the WHOINN together with other nomenclature groups establish new affixes and rules, in order to govern the classification of new substances and to standardize pharmaceutical nomenclature. Unfortunately, these nomenclature rules have not always been observed when naming a new drug. On the other hand, linking the affixes with the groups of the ATC classification system is an important challenge to be met in future work, because the ATC provides a global standard for classifying medical substances and serves as a tool for drug research. In addition, this classification system is also used for reporting ADRs. Recently, Gurulingappa et al. [2009] have developed a system which predicts the ATC class for unclassified drugs by the use of IE and machine learning techniques.

Acronyms, frequently used in biomedical texts to rename drugs and other concepts, have not been addressed in this thesis. The high ambiguity of these terms and the lack of acronym dictionaries make their automatic resolution a difficult task. Nevertheless, it is essential in order to achieve a complete coverage in the drug name recognition process.

Building a manually annotated corpus is a time-consuming, labor-intensive and expensive task. Machine learning methods are not often applied in the biomedical domain because of the shortage of training data. In addition, the major drawback of the biomedical corpora for BNER is that the most of them are limited to protein and gene names. For this reason, the DrugNer corpus could encourage research on automatic extraction information of DDIs, ADRs and other drug information.

Resolving drug acronyms, extending the set of affixes, including additional clues for those affixes that are too short and ambiguous are some challenges for our future research to improve the coverage and the accuracy of drug name recognition and classification tasks. In addition, the affixes could be helpful in the classification of other types of concepts such as organic chemical, enzymes, vitamins and others. Regarding the improvement of the DrugNerTool viewer, we are planning to extend it to allow the end-users themselves to annotate drug names as well as other biomedical concepts, and to correct or update the information provided by DrugNer. We are also planning to link each recognized drug to documents that show additional information such as database identifiers, synonyms, brand names, pharmacological information (drug adverse events, indications, mechanism of action, drug targets, etc), chemical information (structure and formula), and related literature.

Chapter 5

Anaphora Resolution for Drug-Drug Interaction Documents

5.1. Introduction

Information Extraction (IE) techniques can be a useful instrument to manage the knowledge on DDIs. Nevertheless, IE at the sentence level has a limited effect because there are frequent references to previous entities in the discourse, a phenomenon known as *anaphora*.

The extraction of DDIs is a difficult task whose complexity increases when one or both drugs involved in an interaction are expressed with an anaphoric expression, as shown in the following text excerpts taken from the DrugBank database.

Example 7 DDIs expressed by anaphoric expressions

Cimetidine is reported to reduce hepatic metabolism of certain *tricyclic antidepressants*, thereby delaying elimination and increasing steady-state concentrations of *these drugs*.

Triamterene, *metformin* and *amiloride* should be co-administered with care as *they* might increase dofetilide levels.

Figure 5.1: Anaphora resolution can help to improve the performance of the DDI extraction from texts.

All of these examples share a requirement: the need to identify and resolve the anaphoric expressions for detecting the DDIs. The problem of resolving pronominal and nominal anaphora to improve a system that detects DDIs is addressed in this chapter.

Figure 5.2 shows the pipeline architecture of the DDI extraction framework. The text analysis and drug name recognition modules have been described in the chapters 3 and 4 respectively. Briefly, texts are processed by the MMTx program. This tool performs sentence splitting, tokenization, POS-tagging, chunking, and linking of phrases with UMLS concepts. Then, drugs found in such documents are classified into drug families by the DrugNer component. Over this basis, anaphora resolution is carried out to account for both nominal phrases referring to drugs and pronouns. Finally, the output of the previous modules is sent to the relation extraction module that exploits this information in order to account for DDIs in biomedical documents.
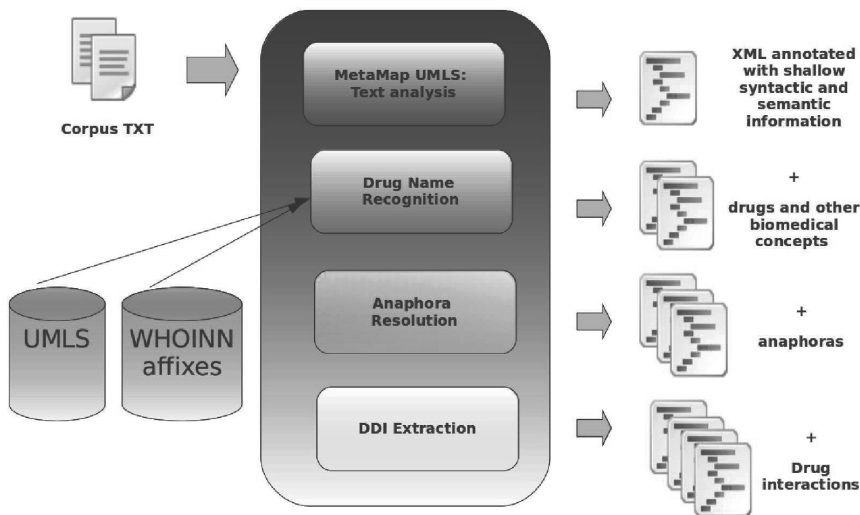
Figure 5.2: DDI Extraction prototype

In Natural Language Processing (NLP) research, two major approaches have been applied to anaphora resolution, heuristics-based and machine learning-based approaches. The former requires to define and apply the different heuristics, constraints or linguistic rules in a predefined order. Contrasting to this approach, machine learning methods do not require the arduous labor of defining rules manually, however, their major drawback is that they usually require a large amount of training data to be effective.

We have defined two different approaches to address the problem of co-referring expressions in pharmacological literature. Besides, as this is the first work that addresses this issue, a corpus has been created in order to analyze the phenomena and evaluate both approaches.

Our first approach is based on a scoring method similar to other works in the biomedical domain [Castano et al., 2002, Lin et al., 2004, Kim and Park, 2004a]. It uses a combination of the domain-specific syntax and semantic information provided by MMTx with generic heuristics. Evaluation shows that this approach achieves results similar to other approaches for anaphora resolution in the biomedical domain.

Our second approach uses a set of linguistic rules inspired by the Centering Theory [Grosz et al., 1995] and constraint satisfaction process over the analysis provided by MMTx. Semantic information from UMLS is also integrated in order to improve the recognition and the resolution of drug nominal anaphora. Evaluation shows how this approach outperforms our first approach and offers an interesting possibility to be developed for other sub-domains in biomedicine. This linguistic rule-based approach shows very promising results for the challenge of accounting for anaphoric expressions in pharmacological texts.

The chapter is organized as follows: Section 5.2 reviews the main approaches addressed for biomedical anaphora resolution. The corpus building for developing and evaluating both approaches is presented in the section 5.3. The detection of anaphoric expressions detection is described in section 5.4. Sections 5.5 and 5.6 describe the scoring-based and centering-based approaches to resolve the antecedent candidates, respectively. Besides, a baseline system has been developed for drug anaphora resolution that is shown in section 5.7. Experimental results of both approaches are shown and compared in section 5.8. Finally, section 5.9 offers some conclusions and some unresolved issues.

5.2. Related Work in biomedical anaphora resolution

Anaphora resolution is often required to improve the results of relation extraction task. It is useful to identify the correct arguments of relations in large sentences with complex structure like subordinate

clauses. Besides, it also helps to extract information beyond the sentence level. It can be found two major approaches in the literature:

1. Heuristic approaches that integrates different knowledge sources like gender and number agreement, syntactic patterns or semantic information to obtain a plausible list of candidates [Poesio et al., 2002, Refoufi, 2006, Wu and Liang, 2009]. The major drawback of these approaches is that it is a very labor-intensive and time-consuming task to construct the domain knowledge base necessary for resolving the anaphora.

2. Machine learning approaches compute the most likely candidate based on previous examples. These approaches can sort out the referred problem in heuristic approaches, however it usually comes across data sparseness problem of language modeling, so they require a large amount of data to train the algorithm [Bunescu, 2003, Ng, 2005].

In the biomedical domain, the lack of available corpora motivated that early approaches were mostly based on heuristics. In this sense, Castano et al. [2002] present a method for resolving anaphoric expressions for candidates taken from MedLine articles and abstracts. By defining a different range of resolution scope for each type of anaphoric expression, it uses different morphological, syntactic and semantic features such as number or semantic type agreement (UMLS typing-based system), longest common subsequence for similarity among candidate antecedents and coercion-type matching (most suitable agent / patient linguistic role according to the verb) from the most frequent bio-relevant verbs in Medline. Each possible antecedent of a certain anaphora was given a different cumulative score according to the significance of its linguistic features and the one with the best salience measure was chosen. General results are 73.8% f-measure over a corpus of 46 MedLine abstracts which were annotated by a domain expert.

Lin et al. [2004] also applied this scoring technique but they restrict the types of nominal anaphoric expressions to be taken into account, enrich the syntactic features with new values and apply coercion-type matching as before, using Genia corpus. General results are 92% f-measure in pronominal anaphora and 78% in nominal anaphora in 32 Medline abstracts (MedStract corpus) [Pustejovsky et al., 2002a]. This approach is improved in [Liang and Lin, 2005] by using new resources like WordNet or PubMed for finding semantic relationships among concepts not found in UMLS. They extend the MedStract corpus with 100 Medline abstracts obtaining 87.43% f-measure for pronominal anaphora and 80.61% for nominal anaphora.

Anaphora resolution applied to the field of PPIs can be found in [Kim and Park, 2004a], which presents an anaphora resolution system integrated in a larger PPIs extraction study, so-called BioAR. It identifies antecedents of pronouns by applying patterns for parallelism and centering theory [Grosz et al., 1995]. Nominal phrase anaphors are identified according to the most salient score, using similar features as in [Castano et al., 2002]. Experimental results are 75% precision and 56.3% recall in pronoun resolution and 75% precision and 52.2% in definite noun phrase resolution from 120 unseen biological interactions extracted by BioIE system[Kim and Park, 2004b].

Likewise, in [Sánchez et al., 2006] the impact of anaphora resolution on the result of a protein interaction extraction system is analyzed by using the Guitar system [Poesio and Kabadjov, 2004] over the 20 full texts and abstracts of the Medstract corpus and three articles taken from the Journal of Biological Chemistry. From the 402 PPIs in the corpus, only 20 were conveyed by an anaphoric expression. Results show 70% recall in anaphora resolution in abstracts and 52.65% in full texts. No data about precision are available. Results suggest small improvements in protein extraction.

Regarding machine learning approaches to anaphora resolution in biomedical documents, Nguyen and Kim [Nguyen and Kim, 2008] carried out a comparative study with three different corpora: MUC [Hirschman and Chinchor, 1997] and ACE [Doddington et al., 2004], accounting for the news domain, and Genia for bio-medical documents. They built a machine learning-based pronoun resolver using a Maximum Entropy ranker model that selects the most likely antecedent candidate from a set of candidates by using a huge set of linguistic features divided into baseline attributes like pronoun type, number, gender, string, distance, etc, (mostly used in other approaches) and innovative features like grammatical roles, semantically most appropriate candidate or context information about the anaphoric pronoun. From the latter group, those that improved baseline for each of the corpus were selected obtaining 79.55% (Genia), 64.61%(ACE), 60.42%(MUC) in success rate, which can be defined as the ratio between the number of successfully resolved anaphors and the number of all anaphors.

Anaphoric expressions are resolved in [Gasperin and Building, 2006] presenting a semi-supervised approach that makes use of rich domain resources such as the FlyBase database for the training of a

gene-name recognizer and the Sequence Ontology[Eilbeck et al., 2005] for semantic tagging. Nominal phrases are identified by the use of the domain-independent parser RASP [Briscoe and Carroll, 2002]. The system looks for the closest antecedent matching the anaphoric expression according to a set of linguistic features. It evaluated against two hand-annotated full papers containing 302 sentences and 314 anaphoric expressions and it. System reaches 58.8% precision and 57.3% recall. A summary of main approaches of biomedical anaphora resolution can be found in table 5.1.

Approach	Description	Corpus	Results
[Castano et al., 2002]	Scoring method	46 abstracts	F=0.74
MedStract [Lin et al., 2004]	Scoring method	32 abstracts	F=0.92 pronominal, F=0.78 nominal
[Kim and Park, 2004a]	Centering theory for pronominal anaphora and scoring method for nominal anaphora	120 biological interactions	F=0.64 pronominal, F=0.59 nominal
[Liang and Lin, 2005]	Scoring method	MedStract + 100 abstracts	F=0.87 pronominal, F=0.80 nominal
[Nguyen and Kim, 2008]	Maximum Entropy ranker model	Genia	Success rate: 79.55%

Table 5.1: Summary of the main approaches of biomedical anaphora resolution

To the best of our knowledge, there are no published works tackling the drug anaphora resolution for the case of pharmacological documents. In this chapter, we describe two different approaches for anaphora resolution that work on pharmacological texts following an heuristic approach for anaphora resolution partially motivated by the lack of a large annotated corpus in this domain. The first approach is based on a scoring method similar to other works in the biomedical domain [Castano et al., 2002, Lin et al., 2004, Kim and Park, 2004a]. The second approach uses a set of linguistic rules inspired by Centering Theory Grosz et al. [1995] and constraint satisfaction. Both approaches use a combination of the domain-specific syntax and semantic information provided by MMTx.

5.3. Building a corpus to support the anaphora reference resolution for Drug-Drug Interactions

There is no existing corpus devoted to anaphoric expressions resolution in pharmacological texts, so a corpus has been built for research and evaluation.

A set of 49 documents from the DrugDDI corpus has randomly been selected annotated manually by a linguist with the assistance of a pharmaceutical expert. The annotation has been made on the output of text analysis and DrugNer modules in XML format.

Each of the documents has on average 40 sentences and 716 words. Table 5.2 shows statistics about the corpus created for evaluation purposes. The third column represents the number of phrases assigned to some of the UMLS semantic types that can represent pharmacological substances. These UMLS semantic types referring to drugs were selected by a pharmacist and are: *Clinical Drug (clnd)*, *Pharmacological Substance (phsu)*, *Antibiotic (antb)*, *Biologically Active Substance (bacs)*, *Chemical Viewed Structurally (chvs)* and *Amino Acid, Peptide, or Protein (aapp)*. A more detailed description of these semantic types can be found in section 3.2.3.

Anaphora is a linguistic procedure to refer to entities that usually come up in the recent discourse (antecedents). Its resolution is essential to understanding the meaning of a certain expression. There are two kinds of anaphora that are prevalent in biomedical literature:

1. **Pronominal anaphora**. In this case, an entity is referred to by a pronoun. The set of more prevalent pronouns was identified in the DDIs corpus: personal (*it, they, reflexive itself, themselves*), relative (*which, that*), distributive (*each, either, neither*) and indefinite (*all, some, many, one*). As this approach focuses on drug interactions those pronouns that could not refer to drug entities such as *I, me, you, your, who*, etc., were ruled out.

2. **Nominal (phrase) anaphora**. This is the case of an entity being referred to by a nominal phrase. This approach focuses on the domain-relevant nominal phrases, that is, those that refer to drugs or

Type of Phrase	Phrases	Drugs
Noun (NP)	4,935	406
Prepositional (PP)	2,157	119
Verbal (VP)	4,347	3
Adjectival (ADJ)	89	1
Adverbial (ADV)	605	0
Conjunctions (CONJ)	1,544	0
Unknown (UNK)	2,535	14
Total :	18,035	689
Total Sentences:		1,975

Table 5.2: Some characteristics of the corpus for anaphora resolution

drug properties in pharmacological documents. These phrases consist of the definite article (*the*), possessives (*its, their*), demonstratives (*this, these, those*), distributives (*both, such, each, either, neither*) or indefinites (*other, another, all*), followed by a generic term for drugs (such as *antibiotic, medicine, medication*, etc) or a drug property or effect. Some examples of drug nominal anaphora are: *the drugs, these anticoagulants, its pharmacological effects, their anticoagulant properties*.

A linguist annotated all anaphors in the corpus and their corresponding antecedents in the XML format, so such linguistic relations could be retrieved automatically. The corpus contains a total of 331 anaphoric expressions. Table 5.3 and Table 5.4 show the distribution of the pronominal and nominal anaphors in the corpus.

Figure 5.3 shows an example of a sentence processed by MMTx and DrugNer and annotated with anaphoric expressions. For each phrase, it is offered its type as well as the CUI, the name and the semantic types of the UMLS concepts provided by MMTx (just in case, the text of the phrase is founded in the UMLS Metathesaurus). Let us take as example the prepositional phrase *with aprazolam* (*s28.p369*) which has been mapped to the UMLS concept *Alprazolam* (CUI=*C0002333*) whose semantic types are *Organic Chemical* (*orch*) and *Pharmacological Substances* (*phsu*). Moreover, DrugNer classified it into the drug family *Benzodiazepine derivative*, by the affix *-azolam*. The example also contains a pronominal anaphoric expression (phrase *'it'* (*s28.p78*)) whose antecedent is annotated by the attribute *ID_ANTENCENT*. In this case, the antecedent is the phrase *of fluvoxamine s28.p71*.

Pronominal Anaphora	Num
Personal (*it, they*)	23
Reflexives (*itself, themselves*)	1
Relatives (*which, that*)	113
Distributive (*both each, either, neither*	8
Demonstrative (*these, this, those, that*)	12
Indefinite (*all, some, many, one*)	8
Total Phrases:	165

Table 5.3: Distribution of pronominal anaphora in the corpus.

Nominal Anaphora	Num
Definite (*the*)	37
Possessive (*its, theirs*)	52
Distributive (*both, each, either, neither*)	11
Demonstrative (*these, this, those, that*)	58
Indefinite (*other, another, all*)	8
Total Phrases:	166

Table 5.4: Distribution of nominal anaphors in the corpus.

```
– <SENTENCE ID="s28" TEXT="Moreover, as noted with alprazolam, the effect of fluvoxamine may
   even be more pronounced when it is administered at higher doses.">
  – <PHRASES>
    + <PHRASE ID="s28.p366" NUMTOKENS="2" TEXT="Moreover" TYPE="UNK"></PHRASE>
    + <PHRASE ID="s28.p367" NUMTOKENS="1" TEXT="as" TYPE="CONJ"></PHRASE>
    + <PHRASE ID="s28.p368" NUMTOKENS="1" TEXT="noted" TYPE="VP"></PHRASE>
    – <PHRASE ID="s28.p369" NUMTOKENS="3" TEXT="with alprazolam" TYPE="PP">
      – <MAPPINGS>
          <MAP CUI="C0002333" NAME="Alprazolam" NAME_SHORT="Alprazolam"
          PROB="1000" SEMTYPES="orch,phsu" STEM_0="-azolam"
          DRUGFAMILY="Benzodiazepine derivatives"/>
        </MAPPINGS>
      + <TOKENS></TOKENS>
      </PHRASE>
    + <PHRASE ID="s28.p370" NUMTOKENS="2" TEXT="the effect" TYPE="NP"></PHRASE>
    + <PHRASE ID="s28.p371" NUMTOKENS="2" TEXT="of fluvoxamine" TYPE="PP/of">
      </PHRASE>
    + <PHRASE ID="s28.p372" NUMTOKENS="1" TEXT="may" TYPE="VP"></PHRASE>
    + <PHRASE ID="s28.p373" NUMTOKENS="1" TEXT="even" TYPE="ADV"></PHRASE>
    + <PHRASE ID="s28.p374" NUMTOKENS="1" TEXT="be" TYPE="V/be"></PHRASE>
    + <PHRASE ID="s28.p375" NUMTOKENS="1" TEXT="more" TYPE="ADV"></PHRASE>
    + <PHRASE ID="s28.p376" NUMTOKENS="1" TEXT="pronounced" TYPE="VP"></PHRASE>
    + <PHRASE ID="s28.p377" NUMTOKENS="1" TEXT="when" TYPE="CONJ"></PHRASE>
    + <PHRASE ID="s28.p378" ID_ANTECEDENT="s28.p371" NUMTOKENS="1" TEXT="it"
       TYPE="NP"></PHRASE>
    + <PHRASE ID="s28.p379" NUMTOKENS="1" TEXT="is" TYPE="V/be"></PHRASE>
    + <PHRASE ID="s28.p380" NUMTOKENS="1" TEXT="administered" TYPE="VP">
      </PHRASE>
    + <PHRASE ID="s28.p381" NUMTOKENS="4" TEXT="at higher doses" TYPE="PP">
      </PHRASE>
    </PHRASES>
  </SENTENCE>
```

Figure 5.3: Example of sentence processed by MMTx and DrugNer and annotated with resolved anaphoric expressions

5.4. Identification of anaphoric expressions

The anaphora resolution issue can be split into two main steps: identification of anaphoric expressions and selection of antecedents. This section addresses the former step that is shared by both of our approaches.

The identification of anaphoric expressions is carried out through several steps of selective filtering. A first filter restricts the type of the phrase by selecting those with type *NP*, *PP* or *UNK* (unknown phrase) as possible candidates. Moreover, a detailed observation of the corpus revealed that MMTx misidentified phrases with *both*, *either*, *neither*, or *each*, annotated as *CONJ* instead of *NP*. Thus, these kinds of phrases are also selected as anaphoric candidates.

5.4.1. Identifying pronominal anaphora

Regarding pronominal anaphora, the module selects those phrases referred to in table 5.3. Singular and plural pronouns in first and second grammatical person are filtered out because they usually refer to other entities (usually patients or health care professionals) rather than drugs. Moreover, the pleonastic-it expressions are excluded by using the rules proposed in Lin et al. [2004]. A pleonastic pronoun refers to an occurrence of a pronoun that can be used without referring to any specific entity. These rules are extended to recognize the negation and modal verbs as possible arguments in this kind of expressions (see table 5.5).

5.4.2. Identifying drug nominal anaphora

In the case of nominal phrase anaphora, the module selects those phrases with determiners or articles in table 5.4. However, it must be borne in mind that anaphora is a linguistic device for referring to previous entities in the discourse and this reference is carried out generically. Drug nominal anaphora are the anaphors that refer to drugs. This is the reason why a semantic restriction based on the semantic type of phrases is used to rule out those phrases that are not classified with some of the UMLS semantic types that represent drugs, since we only tackle the resolution of drug nominal anaphora.

Rules	Examples
IT [MODALVERB [NOT]?]? BE [NOT]? [AJD|ADV|VP]* [THAT|WHETHER]	*It is not known whether* other progestational contraceptives are adequate methods of contraception during acitretin therapy.
IT [MODALVERB [NOT]?]? BE [NOT]? ADJ [FOR NP] TO VP	If *it is not possible to discontinue* the diuretic, the starting dose of trandolapril should be reduced.
IT [MODALVERB [NOT]]? [SEEM|APPEAR|MEAN|FOLLOW] [THAT]*	*It does not appear that* the SSRIs reduce the effectiveness of a mood stabilizer in these populations

Table 5.5: Rules to recognize pleonastic-it expressions.

Thus, candidates are selected if they are attached to a drug family (*these analgesics*, *the oral anticoagulant*, etc) or to a generic term for drugs (such as *this medicine*, *the medication* or *both drugs*). Candidates consisting of specific terms for drugs like *the serum digoxin concentration*, *the warfarin drug*, etc., are disregarded. To achieve this restriction, the approach uses the concept-unique identifier (*CUI*) provided by MMTx to distinguish between phrases linked to abstract or concrete drugs. Therefore, only phrases attached to the concept *pharmacological substance* (CUI=*C1254351*), their direct hyponyms and their hyponym descendants will be selected. Besides, those hyponyms included in the *Medical Entities Dictionary* representing specific terms for drugs were ruled out.

Candidate anaphors consisting of a possessive article have a different semantic restriction. These phrases are usually linked to a combination of several UMLS concepts, called *MultiMap*, one of them representing an abstract drug or drug family and the other representing a property, activity, or an effect of the drug classified with some of the semantic types *Qualitative Concept* or *Activity*. For example, the nominal phrase *with its anti-estrogenic activity* is linked to the combination of two UMLS concepts:

- *Estrogen Antagonists* (CUI=*C0014930*), concept classified with the semantic type *pharmacological substance* (*phsu*). *Estrogen Antagonists* is a drug family.

- *Activity* (CUI=*C0441655*), concept classified with the semantic type *Activity* (*acty*).

A future work is to include additional UMLS semantic types such as *Quantitative Concept*, *Functional Concept* or *Temporal Concept*, in order to provide greater coverage of the drug properties and qualities.

Once a nominal candidate has been selected, it is necessary to determine its grammatical number. Unfortunately, MMTx does not provide this information, so every phrase's head noun was matched against a set of lexical patterns (see table 5.6) to decide its number.

Number	Lexical pattern
Plural:	[A-Z]+(S|IES|OES|XES|SHES|CHES|SES|ZES)
Exception for singular:	[A-Z]+(U|S)S

Table 5.6: Lexical patterns for deciding grammatical number.

Finally, for distinguishing nominal phrases and pronouns consisting of units *both, either, neither* from correlative expressions, a regular expression (see table 5.7) is developed.

Rule	Example
[BOTH|EITHER|NEITHER][NP|PP|UNK] [AND|OR|NOR] [NP|PP|UNK]	These pharmacokinetic effects seen during diltiazem coadministration can result in increased clinical effects (e.g., prolonged sodation)of *both midazolam and triazolam.*

Table 5.7: Regular expression for detecting correlative expressions.

5.5. Scoring-based method for resolving antencedent candidates

The first approach is based on a scoring method similar to other works in the biomedical domain [Castano et al., 2002, Lin et al., 2004, Kim and Park, 2004a]. It uses a combination of the domain-specific syntax and semantic information provided by MMTx with generic heuristics. Once the anaphoric expressions have been identified, the antecedent candidates must be found in the text.

Corpus observation showed that most antecedents usually occur in the previous context of their referring expressions, so only phrases in the range of two sentences are considered. According to this scope, this method selects those phrases whose syntactic type is *NP*, *PP* or *UNK* and semantically classified with some of the UMLS semantic types that represent drugs as possible candidates.

This semantic restriction is not applied for pronominal anaphora resolution since antecedents are not semantically determined by their pronominal anaphoric expressions.

From the resulting list, number agreement between anaphora and its candidate antecedent is checked out. The same lexical patterns (table 5.6) as for the analysis of anaphoric expressions are applied to determine the number of the antecedent.

Besides, a regular expression is applied to detect coordinate structures occurring inside a sentence, that are taken as possible antecedents in plural grammatical number. This pattern is helpful to resolve those anaphors matching plural antecedents if they are expressed by mean of a coordinate structures as shown in table 5.8. In this case, it is necessary to resolve that the pronoun *they* refers to all drugs included in the coordinate structure: *fluoxetine, sertraline and paroxetine*.

Rule	Example
([NP|PP|UNK],)* [NP|PP|UNK] [NP|PP|UNK] [AND|OR|NOR]	While *fluoxetine, sertraline and paroxetine* inhibit P450 2D6, *they* may vary in the extent of inhibition.

Table 5.8: Rule to detect coordinate structures.

Once the list of antecedent candidates is determined, the method assigns a salience measure to each antecedent candidate according to distance. The closer a certain candidate and anaphora are, the more probable it is that the candidate will be selected as the antecedent. The formula used is:

$$distance\ score(candidate_i) = N * [\frac{d_{max} - d_i}{d_{min} - d_i}] \tag{5.1}$$

where

- N is the maximum weight assigned to the distance factor. The experiments determined that the most appropriate value of the parameter N is 3.

- $dmax$ is the distance between the most faraway candidate and anaphora according to the number of phrase elements between them,

- $dmin$ is the distance between the closest candidate and the anaphora, and

- d_i is the distance between the anaphora and the candidate to be evaluated.

In addition to the distance factor, the longest common subsequence shared between the anaphora and the antecedent is also considered. The function for weight assignation according to common morphological subsequence is expressed as follows:

$$morphological\ score(candidate_i) = N * (1 - [\frac{min - LCS(anaphor, candidate_i)}{min}]) \tag{5.2}$$

where

- N is the weight assigned to this factor (N=1.5 in experiments),

- min is determined by selecting the smaller length between the anaphora and the candidate and

- LCS is the shared length between the anaphora and the candidate.

Finally, the sum of both scores is assigned to the candidate. The candidates are ordered by score and only those exceeding the threshold (1.5) were selected; if candidates did not go beyond this value, the anaphora was unresolved.

5.6. Linguistic rules-based method for resolving antecedent candidates

This section explains the second of the two approaches for drug anaphora resolution proposed in this thesis. This approach uses a set of linguistic rules inspired by Centering Theory Grosz et al. [1995] and constraint satisfaction over the analysis provided by MMTx.

In this method, the detection of antecedents issue can be descomposed in two different phases: determination of anaphor scope and selection of antecedents.

5.6.1. Determination of the anaphora scope

The range of searching for a possible antecedent is not unlimited. As referred, this approach makes use of the framework called *Centering* [Grosz et al., 1995] to account for the way information is linguistically structured and focused . Entities (centers) referred to in an utterance serve to link that utterance to others in the segment that contains them. The main claims of this theory applied to anaphora resolution are the following:

1. The choice of a center (antecedent) for a certain anaphora is from the set of entities (centers) of the previous utterance (locality).

2. Entities mentioned in an utterance are more central than others according to the function:

$$subject > object > other \qquad (5.3)$$

3. Each anaphoric expression in an utterance has exactly one antecedent (center).

Thus, based on the third claim, the anaphoric expressions are associated to just one antecedent. This antecedent is taken from the previous ordered sequence of entities (centers), (first and second claims). Basically the method tries to match an anaphoric expression against candidates in the same sentence sorted by position from left to right. The more central an entity is, the higher the possibility it is to be located on the left side of a sentence, since subjects are usually at the beginning. In case no antecedent matches, it moves backward up to the previous sentence and searches for antecedents from left to right again.

However, it was observed that the Centering Theory cannot account for certain types of anaphoric expressions whose antecedents are in most cases to be found locally. Relative, reflexive and possessive anaphoric expressions find their antecedent in the previous context in most of the cases, so it was decided not to apply Centering Theory on this kind of expressions and link them to the closest nominal phrase that satisfied their semantic and morphological restrictions.

5.6.2. Antecedent selection

Finally, for each of the candidates selected in the previous phase, the method checks one by one whether their linguistic features are consistent with features of the anaphoric expression, as follows:

1. Nominal phrases and pronouns have to present number agreement with their antecedents. Nominal phrases in coordinative or appositive relation are taken as the same center (antecedent) by the pattern defined in 5.8.

2. Additionally, nominal phrase anaphors following centering restrictions are determined to match nominal phrases representing drugs, in particular those phrases classified by MMTx according to some of the UMLS semantic types that represent drugs. Likewise, these phrases must not be composed of abstract drugs (drug families or phrases such as *the medicine* or *this drug*), but a drug specifically.
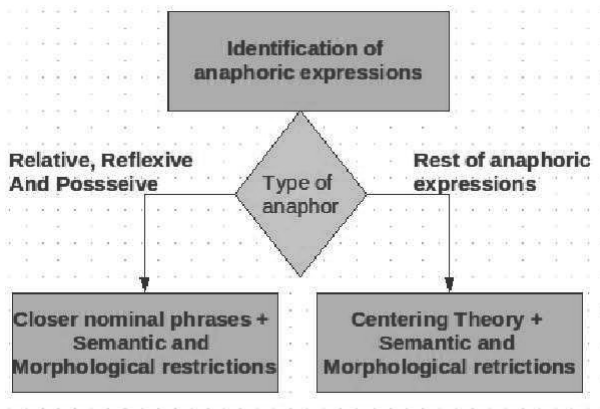
Figure 5.4: Summary of the Linguistic rules-based approach

5.7. A baseline for drug anaphora resolution

As there is no previous work on anaphora resolution in pharmacological texts, it was decided to develop an ad-hoc baseline strategy for anaphora resolution that simply selects the closest nominal phrase. Anaphoric expressions considered are those referred to in tables 5.3 and 5.4 .

For testing the performance of the baseline system, the F-score measure with $\beta = 1$, also called balanced F-score, is used. This is a weighted harmonic mean of precision and recall. Precision is the ratio between the anaphors successfully resolved by the approach and the anaphors proposed by the approach. Recall is the ratio between the anaphors successfully resolved by the approach and the number of anaphors occurring in the corpus. Results obtained for the baseline system are shown in the following tables.

Type	Total	Precision	Recall	F
Personal	23	0.26	0.26	0.26
Reflexive	1	1	1	1
Relative	120	0.83	0.81	0.82
Distributive	8	0.33	0.12	0.18
Demonstrative	11	0	0	0
Indefinite	8	0.25	0.12	0.16
Global results	164	0.67	0.65	0.66

Table 5.9: Baseline for pronominal anaphora resolution.

Total	Total	Precision	Recall	F
Definite	37	0	0	0
Possessive	52	0.53	0.42	0.47
Distributive	11	0.20	0.27	0.23
Demonstrative	58	0.03	0.01	0.02
Indefinite	8	0	0	0
Global results	166	0.23	0.15	0.18

Table 5.10: Baseline for nominal anaphora resolution.

5.8.   Experiment results of the anaphora resolution

This section shows and compares the results obtained with the two methods for drug anaphora resolution. Results of the anaphora resolver are compared to those provided by the manually annotated corpus.

Regarding the scoring-based method, from the 330 anaphoric expressions obtained for the types analyzed in the corpus, 265 are detected by the method and 222 are successfully resolved, that is, attached to the correct antecedent. Results are the following:

Total	Baseline			Scoring			Inc
	Precision	Recall	F-baseline	Precision	Recall	F-approach	
330	0.49	0.40	0.44	0.77	0.62	0.69	0.57

Table 5.11: Global results of the baseline and the scoring-based approach

Type	Total	Baseline			Scoring			Inc
		P	R	F	P	R	F	
Personal	23	0.26	0.26	0.26	0.52	0.52	0.52	1.00
Reflexive	1	1	1	1	1	1	1	0
Relative	120	0.83	0.81	0.82	1	0.92	0.96	0.17
Distributive	8	0.33	0.12	0.18	0.85	0.75	0.8	3.44
Demonstrative	11	0	0	0	0.33	0.09	0.14	∞
Indefinite	8	0.25	0.12	0.16	0.8	0.5	0.61	2.81
Global results	164	0.67	0.65	0.66	0.9	0.82	0.85	0.29

Table 5.12: Results of the scoring-based method for pronominal anaphora resolution.

Type	Total	Baseline			Scoring			Inc
		P	R	F	P	R	F	
Definite	37	0	0	0	0.63	0.37	0.47	∞
Possessive	52	0.53	0.42	0.47	0.67	0.67	0.67	0.43
Distributive	11	0.20	0.27	0.23	0.60	0.54	0.57	1.48
Demonstrative	58	0.03	0.01	0.02	0.53	0.25	0.34	16
Indefinite	8	0	0	0	0.33	0.12	0.34	∞
Global results	166	0.23	0.15	0.18	0.61	0.42	0.50	1.78

Table 5.13: Results of the scoring-based method for nominal anaphora resolution.

The scoring-based approach obtains a 57% relative improvement over the baseline model in overall results. The difference is even more pronounced for the case of nominal anaphora resolution with an increase of 178%. The increment is calculated with the following function:

$$Inc = \frac{F_{approach} - F_{baseline}}{F_{baseline}} \tag{5.4}$$

Clauses in the corpus are characterized by frequent coordinate and subordinate structures, along with numerous prepositional phrases which explain the difficulty of the task and the results of the baseline. From the results it is clear that linguistic information is needed in order to deal with anaphora in this kind of documents. The contribution of semantic resources like MMTx becomes evident when comparing the approach against the baseline.

Pronominal anaphora resolution performs better than its counterpart not only in precision but also in recall. Likewise, the good performance in the resolution of relative pronoun antecedents must be emphasized, explained by the fact that these units are mostly located directly to the left of the anaphoric expressions. In addition, as pointed out in [Poprat and Hahn, 2007], pronominal anaphora is easier to resolve than the nominal one because the latter requires an encyclopedic knowledge source.

Regarding the linguistic rules-based approach, from the 331 anaphoric expressions considered, 265 are detected by the method and 232 are successfully attached to an antecedent. Global results of both

	Baseline			Linguistic Rules			
Total	Precision	Recall	F-baseline	Precision	Recall	F-approach	Inc
331	0.49	0.40	0.44	0.84	0.7	0.76	0.73

Table 5.14: Global results of the baseline and the linguistic rules-based approach

		Baseline			Linguistic Rules			
Type	Total	P	R	F	P	R	F	Inc
Personal	23	0.26	0.26	0.26	0.91	1	0.95	2.65
Reflexive	1	1	1	1	1	1	1	0
Relative	120	0.83	0.81	0.82	1	0.99	0.99	0.21
Distributive	8	0.33	0.12	0.18	0.85	0.87	0.86	3.78
Demonstrative	11	0	0	0	0.33	0.27	0.29	∞
Indefinite	8	0.25	0.12	0.16	0.57	0.62	0.59	2.69
Global results	164	0.67	0.65	0.66	0.92	0.904	0.91	0.38

Table 5.15: Results of Centering-based approach for pronominal anaphora.

		Baseline			Linguistic Rules			
Type	Total	P	R	F	P	R	F	Inc
Definite	37	0	0	0	0.54	0.59	0.56	∞
Possessive	52	0.53	0.42	0.47	0.76	1	0.86	0.83
Distributive	11	0.20	0.27	0.23	0.77	0.90	0.82	2.57
Demonstrative	58	0.03	0.01	0.02	0.81	0.48	0.60	29
Indefinite	8	0	0	0	0.40	0.37	0.38	∞
Global results	166	0.23	0.15	0.18	0.71	0.47	0.56	2.11

Table 5.16: Results of Centering-based approach for nominal anaphora.

baseline and this approach are shown in table 5.14. Results for the different types of anaphora are shown in tables 5.15 and 5.16.

The results obtained by the linguistic rule-based approach achieved an increment of 73% respect to the baseline and outperforms the scoring-based approach for drug anaphora resolution. This is explicable since previous approach emphasized the proximity of the candidate to the anaphoric expression and antecedents can be found at the beginning of the same or previous sentence as it is pointed out by [Grosz et al., 1995].

As it occurred with the scoring-based method, pronominal anaphora resolution performs better than nominal anaphora resolution. Example 8 shows a case in which the centering theory, in particular, its third claim, fails in resolving the drug nominal anaphora *these drugs*.

Based on the third claim of the Centering theory, the anaphora *these drugs* has a only antecedent. That is, only the phrases in plural form or coordinate structures are considered as antecedent candidates. Thus, though *flecainide* and *amioradone* satisfy the semantic restriction (because they are classified as pharmacological substances), they are not selected as antecedent candidates since they are neither singular form nor build a coordinate structure. Then, the method looks for the antecedent from left to right in the previous sentence, and finds the coordinate structure *Quinidine and procainamide doses* that satisfies the number and semantic agreements. However, the correct antecedents for the anaphora are *flecainide* and *amioradone*.

5.9.   Conclusions

Compiling a comprehensive database of DDIs is a relation extraction task that requires the resolution of anaphoric expressions in biomedical and pharmacological texts. It is believed that anaphora resolution would improve the recall of any extraction method and it would be particularly useful for semiautomated compilation of DDIs.

We have defined two approaches for drug anaphora resolution. The first approach is based on a

Example 8 Centering theory fails in resolving the anaphora *these drugs*

Quinidine and procainamide doses should be reduced by one-third when either is administered with amiodarone. Plasma levels of *flecainide* have been reported to increase in the presence of oral *amiodarone*; because of this, the dosage of flecainide should be adjusted when *these drugs* are administered concomitantly.
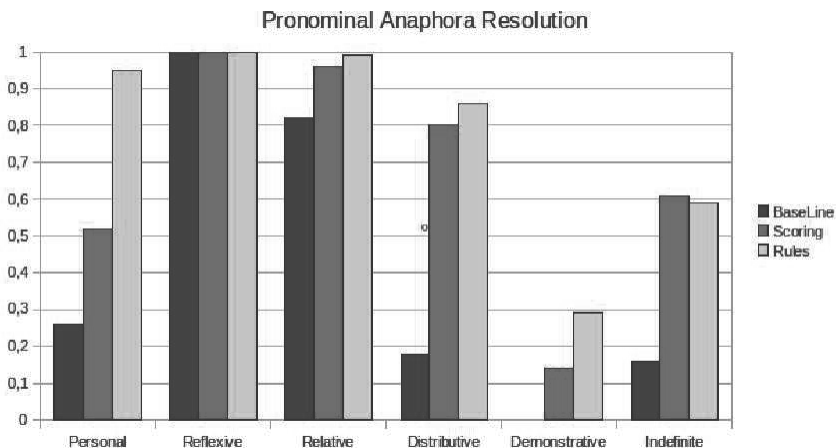
Figure 5.5: Comparasion between results obtained by the three approaches for Pronominal Anaphora Resolution.
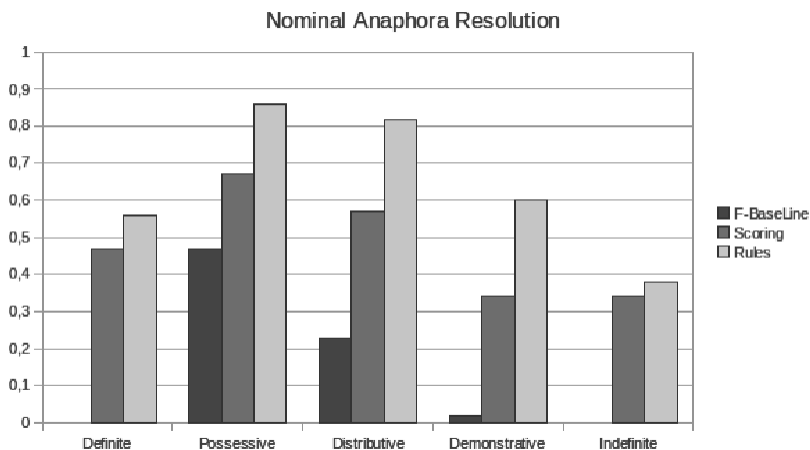
Figure 5.6: Comparasion between results obtained by the three approaches for Nominal Anaphora Resolution.

scoring system and obtains results that are similar to those of other systems referred to in the biomedical domain [Castano et al., 2002, Lin et al., 2004, Kim and Park, 2004a, Liang and Lin, 2005], but it is, to our knowledge, the first research that tackles this issue for the case of DDI documents. It shares with these works the use of a set of linguistic features and a semantic resource. However, it is believed that features weighted in previous approaches [Castano et al., 2002, Lin et al., 2004, Liang and Lin, 2005] such as number agreement constraint must always be satisfied.

The second approach for anaphora resolution uses Centering Theory in order to select the scope of the anaphoric expressions and assign the correct antecedent. In contrast, a simpler heuristic that selects the closer nominal phrase has been experimentally useful in this domain for some types of expressions, relative pronouns and possessive nominal anaphors.

A key component of both approaches is the use of several domain resources, including the MMTx biomedical parser and the UMLS meta-thesaurus. Other approaches that have dealt with biomedical documents have used domain-independent parsers that do not adequately handle the syntactic complexity of biomedical language, including terminology. Unfortunately, MMTx only provides shallow syntactic information, so it can be expected that full syntactic parsing improves the performance of the linguistic rule-based analyzer. UMLS has been useful in order to identify the anaphoric expressions and implement semantic restrictions to candidate resolution.

Our results are not directly comparable to other works, but partially:

1. Syntax changes from a domain to another. Most approaches in the biomedical domain deal with documents from MedLine accounting for any biomedical topic, whereas our documents focus on DDIs. Subsequently, we consider that language style of our documents must be linguistically oriented to the reflection of such relations. Only works [Kim and Park, 2004a] and [Sánchez et al., 2006] deal with documents accounting for PPIs.

2. Works mostly address the anaphora resolution issue by using a set of morphosyntactic properties, so resolution is going to be determined by the way that a document has been analyzed. For example, expressions like *these drugs* or *this medication* are required to be analyzed by a knowledge resource that identifies and analyze them both syntactically (they are nominal phrases in the subject, object or other type of position in the sentence) and semantically (they stand for drugs). Some approaches make use of open-domain analyzers like RASP [Gasperin and Building, 2006].

3. Conversely, other approaches make use of the Genia corpus that has been manually tagged and it does not contain annotation errors (this has a definite influence over results). The degree of precision in annotation is extremely important since results depend on such results. Both approaches presented in this chapter make used of MMTx, that although has shown to be useful for the analysis of biomedical texts, has several syntactic and semantic parsing failures [Divita et al., 2004] that influence negatively the results of both approaches.

To our opinion, from the list of approaches reviewed in this chapter, [Kim and Park, 2004a] is close to both approaches proposed in this chapter. As discussed, such an approach addresses the issue of anaphora resolution in the domain of PPIs, has developed an ad-hoc tool called BioIE to deal with morphosyntactic complexity of this kind of documents and resolution problems have been faced with an approach that also used Centering Theory. As table 5.1 shows our work obtains similar results to [Kim and Park, 2004a] for nominal phrase anaphora resolution and better results for pronominal anaphora.

Future work will consider the overall contribution of the anaphora resolution approaches to the broader task of DDI extraction. Sánchez et al. [2006] have shown that the impact of anaphora resolution on the result of a protein interaction extraction system is very slightly. The DrugDDI corpus is only annotated at sentence level. Although, a pronominal anaphora usually refers to an expression in its same sentence, in the case of a nominal anaphora, the antecedent usually occurs in previous sentences. Thus, we believe that the real contribution of the anaphora resolution can only be measured if the DDIs are annotated at document level. Therefore, we are planning to annotate the DrugDDI corpus at document level, and then, to evaluate the contribution of the anaphora resolution task.

Although sources providing information on interactions such as Medline abstracts and DrugBank may share a common literary style, the distribution of interactions is very different and it also deserves investigation.

Additional extensions of this work include to extend the coverage of the approach to other kinds of biomedical entities (such as genes, diseases or drug targets), the increasing of the size of the corpus in order to make more reliable conclusions, and the application of machine learning techniques that have been successfully applied in other domains. Moreover, semantic information about drug families provided by the DrugNer can be valuable in the resolution of certain nominal anaphors (see example 9).

Example 9 DrugNer could classifies *venlafaxine* like a antidepressant drug, and this information can help to correctly resolve the anaphor *the antidepressant effect*.

Coadministration of naloxone with *venlafaxine* did not modify *the antidepressant effect*

Chapter 6

Related work for Relation Extraction in the biomedical domain

6.1. Introduction

The goal of the relation extraction task is to detect semantic relations between entities in text. This task usually forms part of some application or pipeline processes to support other systems such as Information Retrieval or Question Answering systems in different domains. In particular in the biomedical domain, relation extraction can be used to discover relevant relationships such as PPIs or DDIs. Example 10 shows some types of relations in different domains.

Example 10 Examples of relationships in various domains

born-in: *[Zapatero]$_{PERSON}$ was born in [Valladolid]$_{LOCATION}$*

protein-protein interaction: *[HOX11]$_{PROTEIN}$ interacts with [protein phosphatases PP2A]$_{PROTEIN}$*

drug-drug interaction: *[Fluvoxamine]$_{DRUG}$ given with [warfarin]$_{DRUG}$ may increase the possibility of bleeding.*

food-drug interaction: *Consumption of [grapefruit juice]$_{FOOD}$ increases the plasma concentration of [terfenadine]$_{DRUG}$.*

Although, in general, relationships can involve three or more entities, most of the existing approaches for relation extraction have focused on the extraction of binary relationships. [Sarawagi, 2007] proposes three different scenarios in which the binary relation extraction can be posed: (1) given two annotated entities in an input text, the goal is to find out the type of relationship between both entities, (2) given a relationship R and an entity name E, the goal is to extract all entities that have the relationship R with E in an input text, and (3) given a set of predefined relationships, the goal is to detect all occurrences of those relationships in texts.

Relationship extraction is a complex task that requires to integrate different levels of linguistic processing such as tokenization, PoS tagging, syntactic and semantic sentence parsing. A detailed review of the most common types of resources that are useful for the relation extraction task is presented in [Sarawagi, 2007]. We summarize some of those clues:

- *Context information*, that is, the tokens around and between the two entities. For example, a *DDI* is strongly indicated by the presence of the bigram *interact with* between the two drug names.

- *Part of speech tags* are very useful to identify the entities (which are typically nouns and noun phrases) and verbs (which are crucial to defining the relationship between entities).

- *Full Syntactic parse trees* provide more valuable information than POS tags because they group words in phrase types such as noun, prepositional or verb phrases, which help to understand the relationship between the entities. For example, figure 6.1 shows a parse tree in which the three coordinating noun phrases: *nisindione (NP$_5$)*, *dicumarol (NP$_6$)* and *warfarin* (NP$_7$), and the conjunction *and (CC)* are grouped into the upper noun phrase *(NP$_3$)*. This facilitates to find out the interaction between the aforesaid drugs and *Propylthiouracil*. However, syntactic parsing is a very costly and time-consuming task.

- *Dependency graphs* are an alternative of parse trees. A dependency tree represents the grammatical relations between words in a sentence, so its dependency graph links each word to the words that depend on it.

Example 11 Context Information useful to detect DDIs and adverse drug reactions (ADRs): the underlined words in the below sentences can be useful clues to detect the relationships between the marked entities

(1)[Aspirin]$_{DRUG}$ can <u>interact with</u> [Heparin]$_{DRUG}$
(2)[Aspirin]$_{DRUG}$ may <u>increase</u> serum levels of [methotrexate]$_{DRUG}$
(3)[Aspirin]$_{DRUG}$ may increase the <u>risk</u> of [gastrointestinal bleeding]$_{SYMPTOM}$
(4)[Beta-blockers]$_{DRUG}$ taken for heart disease or high blood pressure can <u>worsen</u> [asthma]$_{DISEASE}$

Example 12 A more reliable extraction of the *interaction* relationship between the below drugs is possible if the word *interact* is tagged as a verb instead of a noun.

(1)Aspirin$_{/noun}$ can$_{/verb}$ interact$_{/verb}$ with$_{prep}$ Heparin$_{noun}$
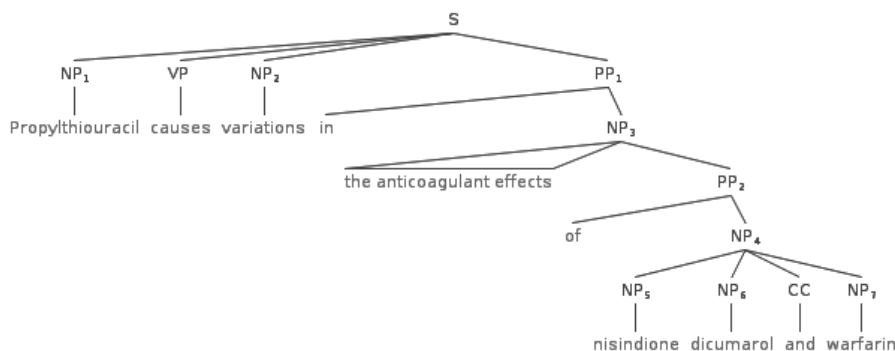
Figure 6.1: Example of parse tree

At it can be seen from the above examples, each of the levels of processing yields to a particular structural representation of its output. For instance, while a sequence of tags is provided in the levels of tokenization and PoS tagging, the structural representation yielded by the syntactic parsing level is a parse tree, and a graph in the dependency parsing level. Handling the diversity of the different representations is a difficult challenge [Sarawagi, 2007].

The goal of biomedical relation extraction is to detect occurrences of a predefined type of relationship between a pair of given entity types. These relationships may be very general such as any biochemical association or very specific such as protein interactions or pharmacokinetic interactions between drugs. Typically, the outcomes of this task are stored in databases, which can be either consulted directly by the users or exploited by data mining algorithms to infer new knowledge. Although in the last decade, this task has received much attention, it is very difficult to compare the results obtained by the different research groups, not only because they use different datasets, but also because they deal with different types of relations. Thus, the performance of this task depends on the type of relationship to be extracted and literature corpus to be processed. Most investigation has centered around biological relationships (genetic and protein interactions) due mainly to the availability of annotated corpora in the biological domain, a fact that facilitates the evaluation of different approaches.

This chapter presents a review on the existing approaches to extract biomedical relations from texts. In general, current approaches can be divided into three main categories: linguistic-based, pattern-based and machine learning-based approaches. We now briefly describe the main categories as follows:

1. *Linguistic-based approaches.* The general idea of these approaches is to employ linguistic technology such as parsing techniques to grasp syntactic structures or semantic meanings that could be helpful to discover relations from unstructured texts. These methods can be further classified into two types, based on the complexity of the linguistics methods, as *shallow parsing* or *deep parsing*.

2. *Pattern-based approaches.* These methods design a set of domain-specific rules (also called patterns) that encode and capture the various forms of expressing a given relationship.

3. *Machine Learning-based approaches.* As opposed to the previous approaches which need a laborious effort to define a set of rules or grammars, machine learning methods allow to automatically acquire and code all the necessary knowledge. Machine learning approaches can be further classified into two types, based on the kind of instance representation:

    a) *Feature-based methods.* These methods extract a flat set of features from the input and represent each instance as a feature vector. Then, a classifier (such as a decision tree or a SVM) is trained using these data instances.

    b) *Kernel-based methods.* Relation instances are encoded as structural representations such as bag of words, word sequence, parse tree or dependency graph. Then, a kernel function must be designed to capture the similarity between structures.

However, many of the existing works adopt hybrid approaches to overcome the difficulties and benefit from the advantages of using each approach. In particular, linguistic techniques such as tokenization, PoS tagging and syntactic parsing, are widely used by both pattern-based and machine learning-based approaches. In the following sections, we present the main works for each category in more detail. In the last section some unsolved problems are presented.

6.2. Linguistic-based approaches

Linguistic-based approaches employ different linguistic techniques to obtain useful information for discovering relations from unstructured texts. Based on the complexity of these techniques, we categorize them into two types: shallow parsing and deep parsing.

Shallow parsing techniques aim to retrieve syntactic information efficiently and reliably from texts, by sacrificing completeness and depth of analysis. The shallow parser EngCG [Voutilainen and Heikkila, 1993] was used by Sekimizu et al. [1998] to obtain morphological and syntactic information in order to detect the subjects and objects of the verbs expressing interactions between proteins. Pustejovsky et al. [2002c] built a cascaded finite state automata to recognize *inhibit* relations. A set of 500 abstracts was manually annotated by experts in biology. Results showed a precision of 90% and a recall of 57%. Leroy et al. [2003] used a shallow parser based on four cascades finite state automates to structure the relations between individual entities. These automates are based on closed-class English words and model generic relations not limited to specific words. The parser can also recognize coordinating conjunctions and captures negation in text. Three cancer researchers evaluated 330 relations extracted from 26 abstracts of interest to them. There were 296 relations correctly detected from the abstracts resulting in 90% precision of the relations and an average of 11 correct relations per abstract.

The systems based on deep parsing deal with the entire sentence structure and therefore are potentially more accurate. [Temkin and Gilder, 2003] used a lexical analyzer and a general context-free grammar (GFG) to extract protein, gene and small molecule interactions from unstructured text. Specific structures of the domain are carried out by the grammar, significantly reducing the complexities of natural language processing. The system achieves a recall rate of 63.9% and a precision rate of 70.2%.

In [Rinaldi et al., 2004], a probabilistic dependency parser is used to identify interactions between genes and proteins. The parser uses a hand-written grammar combined with a statistical language model that calculates lexicalized attachment probabilities. Later, Rinaldi et al. [2007] have also employed a probabilistic dependency parser, Pro3Gres [Schneider et al., 2007], to output functional dependency structures. Based on these structures, interactions between proteins and genes were extracted. Experiments are conducted on two different corpora: Genia and ATCR (which consists of 147 abstracts automatically annotated using the Biolab Experiment Assistant (BEA) tool[1]). Precision values range from 52% to 90% and recall values range from 40% to 60%.

[Fundel et al., 2007b] have developed the RelEx system based on the dependency parse trees to extract relations in biomedical texts. It is applied to one million abstracts to extract gene and protein relations.

---

[1]http://www.biovista.com/bea/main.php?pid=111

About 150,000 relations were extracted with an estimated performance of both 80% precision and 80% recall.

BioPPIExtractor [Yang et al., 2009b] is a PPI extraction system for biomedical literature. This system applies CRF model to tag protein names in biomedical texts, then uses a link grammar parser to identify the syntactic roles in sentences and at last extracts complete interactions by analyzing the matching contents of syntactic roles and their linguistically significant combinations. Link grammar parsers are similar to dependency parsers, but they do not include the head-dependent relationship. In addition, the system includes an anaphora resolution module that focuses on third person pronouns and reflexives. The system was evaluated on the DIP corpus achieving a precision of 55.41%, a recall of namely 41.62%, and an f-measure of 47.53%.

System	Approach	Relation	Corpus	Results
[Pustejovsky et al., 2002c]	Shallow parsing	*inhibit* relations	500 abstracts	P=90%
[Leroy et al., 2003]	Shallow parsing	biological relations	26 abstracts containing 330 relations	P=90%
[Temkin and Gilder, 2003]	Context-free grammar	PPIs	100 abstracts	F1=66.9%
[Rinaldi et al., 2007]	Probabilistic dependency parser	PPIs	Genia, ATCR	P=52-90%, R=40-60%
RelEx	dependency parse trees	PPIs	One million abstracts	F1=80%
BioPPIExtractor	CRF model + Link grammar parser	PPIs	DIP	F1=47.53%

Table 6.1: Main linguistic-based approaches for biomedical relation extraction

The comparison among the different works is not possible because many of them have been evaluated on different corpora. Therefore, it is risky to draw conclusions on the performance of the different techniques. In general terms, the shallow parsing-based approaches perform well for capturing relatively simple binary relationships between entities in a sentence, but fail to extract more complex relationships expressed in various coordinate and relational clauses [Zhou and He, 2007]. Furthermore, many of the shallow parsing-based approaches lack an adequate and rigorous evaluation. Deep parsing-based methods seem to achieve better performance than shallow parsing techniques by analyzing the whole structure of sentences. Among the reviewed systems, RelEx shows the best results with a significant difference between them and those reported by Rinaldi et al. [2007] or the BioPPIExtractor system, however they are all based on dependency parsing. We believe that the performance of these approaches is strongly influenced by the shortage of biomedical parsers. General purpose parsers, which have been trained on generic newswire texts, are not able to deal with the complexity of the biomedical sentences that tend to cause problems due to their long length and high degree of ambiguity [Siddharthan, 2006]. On the other hand, analyzing the whole sentence structure incurs higher computational and time complexity.

6.3. Pattern-based approaches

Similar to linguistic-based methods, pattern-based approaches can make use of syntactic information to achieve better performance, although it can also work without prior parsing and tagging of the text.

Ono et al. [2001] present a system for extracting information on PPIs from the scientific literature. They manually defined A total of 20 patterns to extract PPIs. The system only employs a protein name dictionary, surface clues on word patterns and simple part-of-speech rules. Negation structures are also tackled. The system achieves high recall and precision rates for *yeast* (recall=86.6% and precision=94.3%) and *Eschirichia coll* (recall=82.5% and precision=93.5%).

The SUISEKI [Blaschke and Valencia, 2002] system uses a set o rules based on morphological, syntactical, and contextual information to detect gene and protein names and interactions in scientific texts. Sentence negations and the distance between two protein or gene names are also considered. A probability score is induced for each predefined rule depending on its reliability and used as a clue to score the interaction events. They used a collection of almost 50.000 abstract to build the system, but only 100 sentences were used to evaluate the precision and 100 abstract to estimate the recall. The system achieves a recall of 68.7% and a precision of 44.9%.

The system BioRAT [Corney et al., 2004] focuses on the extraction of PPIs and is able to analyze both abstracts and full-lenght papers. Protein names are recognized using a set of gazetteers that are compiled from sources such as MeSH, Swiss-Prot [Boeckmann et al., 2003] and other hand-made lists. A set of lexico-semantic patterns is defined manually to identify PPIs. To evaluate the system, they collected 394 interactions from the DIP database. These interactions correspond to 229 abstracts. The dataset is called as DIP corpus. Overall, BioRAT achieves 39% recall with 48% precision.

The system *IntEx* [Ahmed et al., 2005] has been developed to extract gene and protein interactions from biomedical texts. Gene and protein names are tagged by the combination of several biomedical terminological resources such as UMLS, GeneOntology and LocusLink database, and a set of regular expressions is also used to recognize the names not found in the resources. Pronoun resolution is also tackled by a very simple heuristic based on the proximity and the number of the noun phrases. Then, the system uses a link grammar to split complex sentences into simple clausal structures made up of syntactic roles. Finally, interactions are extracted from simple sentence clauses by syntactic role matching using a basic set of patterns. Experiments are performed on the DIP dataset. This dataset was created for the evaluation of the BioRAT system [Corney et al., 2004], which is described in the following section. IntExt overcomes the results of BioRAT, reaching a precision of 65.66% and a recall of 26.94%.

Few proposals have tackled the treatment of the negation phenomena and modality [Morante and Daelemans, 2009], however these phenomena are crucial for the correct interpretation of relationships. Negative sentences may also contain evidence of use to biologists and healthcare professionals, and speculative sentences should also be presented with lower confidence. The systems NegExt[Chapman et al., 2001] and Lexer [Mutalik et al., 2001] are based on the use of regular expressions to identify negative PPIs that are expressed with explicit negative particles such as *no* and adverbial *not*. [Sánchez-Graillet and Poesio, 2007] have developed a heuristic-based system to extract negation of PPIs by the use of affixes and a set of inherently negative words. Kim et al. [2006] have focused on the extraction of contrastive relations such as *but not*.

Hand-built pattern-based approaches achieve good performance. However, it is essential that domain experts get involved in the definition of the patterns. This task requires labor-intensive manual processing. On the other hand, these patterns cannot easily adapt to other subdomains. Several approaches have addressed these shortcomings by automatically learning rules or patterns from texts.

In [Phuong et al., 2003], sentences are parsed by a link grammar parser and used to learn extraction rules automatically from a set of seed hand-tagged sentences. Interactions are detected by using heuristic rules based on morphological clues and domain specific knowledge. A set of 550 sentences was compiled to perform the experiments. The system is evaluated on a set of 550 sentences, achieving a precision of 87% and a recall of 60%.

Huang et al. [2004a] propose to use a dynamic programming algorithm to compute distinguishing patterns by aligning relevant sentences and key verbs that describe PPIs. Then, these automatically constructed patterns are used to identify PPIs by a matching algorithm. A set of 1563 sentences is used to learn patterns and 354 sentences are used to test the matching algorithm. The system achieves a precision rate of 80.5% and recall rate of 80.0%. An extended approach is presented in [Hao et al., 2005] whose goal is to improve the patterns. This second approach designs a minimum description length (MDL)-based pattern-optimization algorithm to reduce and merge patterns. Several experiments were performed on 963 sentences in which 1435 interactions were manually detected. This approach achieves better precision (85.1%) than the previous one, but a lower recall (55.8%) (f-measure of 67.40%). Later, Huang et al. [2006a] proposed a hybrid approach based on the combination of shallow parsing and pattern matching, to extract PPIs from scientific biomedical texts. Shallow syntactic and semantic information is used to resolve appositions and coordinate structures. Clause splitting based on a set of rules is also applied to obtain relative clauses. Thus, long sentences are split into sub-ones, from which relations are extracted by a pattern matching algorithm. The patterns are automatically generated using the algorithm proposed in [Hao et al., 2005]. A collection of 920 sentences was manually annotated by experts, detecting 1423 relationships. The approach achieves an average f-score of 80% on individual verbs, and 66% on all verbs.

A common characteristic of the majority of the participating systems at the the *Interaction Pair Task (IPT)* in the BioCreative challenges [Krallinger et al., 2008, 2009] is the usage of pattern matching techniques. In the BioCreative II challenge, the best results were achieved by the system presented in [Huang et al., 2008]. This system, firstly, filters out the articles that are irrelevant. Protein and organisms are recognized by using the databases SwissProt and the NCBI taxonomy [NCBI], respectively. Every protein pair is viewed as an interaction candidate. Then, a set of patterns is generated using the semi-supervised method for learning patterns proposed in [Ding et al., 2007]. The system was trained on a

corpus of 740 full articles and evaluated on a test set of 358 articles, showing a precision of 38.9%, a recall of 30.7% and an f-measure of 28.9%, which reflects the complexity of the task. In the last BioCreative II.5 challenge, most of the participating systems were also based on the use of pattern matching methods to extract PPIs [Verspoora et al., 2009, Hakenberg1a et al., 2009, Sætre et al., 2009]. Sætre et al. [2009] have developed a system for relation extraction called AkaneRe, which has been applied to the IPT task. Texts are analyzed with the Genia-tagger [Tsuruoka et al., 2005] and the Genia Dependency parser [Sagae, 2007]. The protein names are recognized by a maximum entropy markov model trained on the corpus Genia. Then, the system generates templates which are grouped by clustering. The system achieved a precision of 18%, a recall of 27% and an f-measure of 17%. The same system was also evaluated in the BioNLP shared task [Kim et al., 2009], showing the following scores: a precision of 54%, a recall of 28% and f-measure of 37%. Verspoora et al. [2009] use a set of hand-built patterns and define a semantic grammar to detect interactions among the co-occurrences of the proteins. This system achieved a precision of 33.3%, a recall of 22.4% and an f-measure of 25.2%.

In the pharmacological domain, Kolarik et al. [2007] propose to use the lexico-syntactic patterns for identifying and extracting relevant information on drug properties. The goal of the system is to support database content update by providing additionally drug descriptions of pharmacological effects not yet found in databases like DrugBank. The experiments focus on finding out drug families in texts. The system was evaluated on texts from MedLine and from the database DrugBank. The evaluation shows that phrases could be identified with a high performance in DrugBank texts (F-score=89%) and in Medline abstracts (F-score=83%). The evaluation of terms extracted from Medline shows that 29-53% of them are new valid drug property terms. Thus, they could be assigned to existing and new drug property classes not provided by the DrugBank database. Pharmspresso [Garten and Altman, 2009] is a text analysis tool to extract pharmacogenomic concepts and gene-drug interactions from full text articles. An ontology was manually defined with concepts and relationships of interest in the biological domain, in particular, human genes, polymorphisms, drugs and diseases and their relationships. Then, the Textpresso tool [Muller et al., 2004a], which is a template-based text search engine, uses the concepts and relations in the ontology to build regular expressions and look for the templated relationships in text. This tool analyzes texts to find references to human genes, polymorphisms, drugs and diseases and their relationships. A gold-standard of 45 hand-annotated articles was used to evaluate the tool, which identified 78%, 61%, and 74% of target gene, polymorphism, and drug concepts, respectively.

System	Approach	Relation	Corpus	Results
[Ono et al., 2001]	Lexico and syntactic patterns	PPIs	–	F1=87.6-90.2%
SUISEKI	Morpho-syntactic patterns	PPIs	100 sentences	F1=54.3%
BioRAT	lexico-semantic patterns	PPIs	DIP	F1=43.03%
IntEx	Link grammar+patterns	PPIs	DIP dataset	F1=38.9%
[Phuong et al., 2003]	link grammar parser + pattern learning	PPIs	550 sentences	F1=71.02%
[Hao et al., 2005]	pattern learning	PPIs	1435 PPIs in 963 sentences	F1=67.40%
[Huang et al., 2006a]	shallow parsing and pattern matching	PPIs	1423 PPIs in 920 sentences	F1=66%
[Huang et al., 2008]	semi-supervised pattern learning	PPIs	BioCreative dataset	F1=28.9%
AkanePPI	dependency parsing, pattern matching	PPIs	BioCreative dataset	F1=19%
[Verspoora et al., 2009]	semantic grammar + pattern pattern matching	PPIs	BioCreative dataset	F1=25.2%

Table 6.2: Main pattern-based approaches for biomedical relation extraction

As it happens in linguistic-based approaches, pattern-based approaches are not comparable since their experiments are performed on different corpora. Most of the previous works developed and annotated themselves their corpora. It is very striking the difference between the results presented in the BioCreative challenges and those reported by the other works. This may be due to the facts that the texts

in the BioCreative datasets are more complex than those used in the other approaches. Pattern-based approaches usually achieve high precision, but low recall. They are not capable of handling long and complex sentences, so common in biomedical texts. There are several linguistic phenomena including negation, modality and mood, which can alter or even reverse the meaning of the sentence, however, they are not addressed by the pattern-based approaches. Furthermore, these approaches are also limited by the extent of the patterns, since interaction descriptions that span several sentences cannot be detected by them. Thus, these approaches are not able to correctly process anything other than short and straightforward sentences [Zhou and He, 2007], which, on the other hand, are quite rare in biomedical texts.

6.4. Machine learning approaches

Machine learning methods for biomedical relation extraction have gained growing interest in recent years due to their good results achieved in general domain. On the contrary to the pattern-based methods, machine learning approaches do not require to manually encode relevant knowledge, but they formulate the relation extraction problem as a classification task and employ learning algorithms to automatically extract knowledge from texts. Examples of machine learning algorithms are support vector machines, neural networks, k-nearest neighbor algorithm, hidden markov models, and naïve Bayes. In addition, these approaches can be easily extended to new set of data or a new task or domain. However, one major drawback of these algorithms is that they generally require computationally expensive training and testing on large amounts of annotated data. Positive and negative examples must be represented in a suitable format in order to train an algorithm. We can distinguish, depending on the kind of representation, two categories: feature-based and kernel-based methods.

6.4.1. Features-based methods for Relation Extraction

In this subsection, we study the main characteristics of feature-based approaches and review the most relevant works in this field. These methods extract a flat set of features from the input and represent each instance as a feature vector. The feature vectors are used to train an algorithm. Features are extracted from sentences by the application of text analysis techniques including tokenization, POS tagging, shallow or deep parsing, named entity recognition, among others. Sarawagi [2007] provides a review on how the features can be extracted from the different levels of linguistic processing. The feature set should be complete, that is, it must include all features potentially useful for the classification problem. Features can be classified into two different categories: (1) properties of a single token including entity type, PoS tag, lemma and other attributes of the tokens, and (2) relations between tokens: sequence, syntactic or dependency relations between tokens. Typically, the most used features are:

- Words between entities (including themselves).

- Types of entities (person, location, gen, protein, etc)

- Number of words between both entities

- Syntactic parse tree of a sentence can offer more complex and discriminative features. A common feature is the syntactic path between the two entities in a parse tree, or a subtree.

- Dependency graph of the relation instance, that is, information on dependencies among the words in it.

- Number of entities between the two entities, whether both entities belong to same chunk

The system proposed in Rosario and Hearst [2004a] deals with the recognition of entities such as *treatment* and *disease*, and the classification of seven relationships types that can occur between both entities. Five generative graphical models and a neural network were designed using lexical, syntactic, and semantic features from MeSH vocabulary. The experiments show that the MeSH concepts help achieve high classification accuracy.

Xiao et al. [2005] propose a Maximum Entropy (ME) method to extract PPIs from texts. The system combines lexical (such as surrounding words, key words and abbreviations), syntactic and semantic features. Experiments were performed on the IEPA corpus [Ding et al., 2002] (see chapter 3) achieving a recall of 93.9% and a precision of 88.0%.

A system for extracting disease-gene relations from MedLine is described in [Chun et al., 2006]. The system recognizes the disease and gene names and selects the sentences that contain at least one pair of disease and gene names. A dictionary for disease and gene names from six public databases, including HUGO [Povey et al., 2001], RefSeq and LocusLink [Pruitt and Maglott, 2001], Swiss-Prot, DDBJ [Miyazaki et al., 2004] and UMLS [Bodenreider, 2004], is constructed and the relation candidates are extracted by dictionary matching. To build training and testing sets, around one million of abstracts were collected from MedLine. They manually checked 1,000 co-occurrences of gene-diseases. The dictionary matching method achieves a precision of 51.8% and a recall of 100%. To filter out false positives introduced by the dictionary matching process, they used a maximum entropy method to recognize disease and genes names, which achieves to improve the precision to 78.5%, with a reduction of 13% in recall.

A hybrid method that combines dependency parsing and machine learning algorithms is presented in [Katrenko and Adriaans, 2007]. The experiments were performed on two different datasets: the Aimed corpus [Bunescu et al., 2005] and LLL corpus [Nedellec, 2005]. The LLL corpus already consists of the tokenized and parsed sentences provided by LinkParser [Sleator and Temperley, 1995]. The Aimed corpus was parsed using the parser MiniPar [Lin, 1999] providing the dependency parser for each sentence. This method assumes that the entities have already been identified. The set of features includes information on arguments such as lemmas, syntactic functions, information on their parents and the direct ancestor for both arguments. They performed several experiments using three different classifiers: Naïve Bayes, BayesNet and K-nearest neighbor. Precision ranges between 56% and 81% and recall between 32% and 76%, depending on the corpus and classifier used. The best performance is achieved on the Aimed corpus by the combination of the three classifiers (f-measure 72.7%). In general, the three classifiers obtained the worst results on the corpus LLL.

BioPPISVMExtractor [Yang et al., 2009a] is a system for PPIs extraction based on the SVM algorithm. Sentences are parsed using the link grammar parser developed by Grinberg et al. [1995]. The set of features includes surface word, keyword, protein name distance and link path features. The system is trained on the IEAP corpus and tested on the DIP corpus [Corney et al., 2004] (which was developed to test the BioRAT system). The system achieves a recall of 70.04%, a precision of 49.28% and an f-measure of 57.85%.

In the clinical domain, Angus et al. [2008] apply SVM to detect clinically important relationships (such as *has finding*, *has indication*, *has location*, *has target*, among others). The algorithm is trained and tested on a corpus of 77 patient narratives which were manually annotated by two clinically trained annotators. The system achieves an average f-measure of 72%.

In the pharmacological domain, Duda et al. [2005] have evaluated the classification capability of SVM as a method for locating articles about DDIs. They manually created a corpus of 2000 abstracts of positive and negative drug interaction citations. The set of features used to train the SVM model is composed of MeSH terms, CUI-tagged title and abstract text, and stemmed text words. The study shows that automated classification techniques have the potential to perform at least as well as PubMed in identifying DDI articles. Another approach with a similar purpose has been developed in [Rubin et al., 2005]. The goal is to develop an automated method to identify articles in Medline citations that contain pharmacogenetics data pertaining to gene-drug relationships. Three types of statistical models and a heuristic method (a 'gene-drug filter') are implemented to detect pharmacogenetics articles. The statistical models include Naïve Bayes, logistic regression, and a log-likelihood method. A sampling of the articles identified from scanning Medline was reviewed by a pharmacologist to assess the performance of the method. The system achieves an f-measure of 88% with a precision of 80% and a recall of 97%.

Most works have focused on the extraction of PPIs, however, it is not possible to draw precise conclusions about their performance, because many of them have been evaluated on different corpora. In this regard, the BioCreative challenges play a very important role in making possible a realistic evaluation of the different PPIs extraction approaches. In the last BioCreative II.5 challenge, the best results were reported by a system Chen et al. [2009] based on SVM classifier. The feature set used to train the SVM classifier includes the interaction proteins and their context words, the distance in tokens between the two proteins, the number of other recognized proteins between the two interacting proteins, interaction keywords, the position of interaction keywords and the distance in words between the interaction keyword and the protein nearest to it. They also used 16 syntactic pattern features and 2 boolean features to indicate if this interaction pair exists in MINT and IntAct databases. The system achieved a precision of 63.7%, a recall 61.5% and an f-measure of 57.8%. These results are significantly superior to those obtained by other participating systems, which are based on the use of pattern matching (the second-ranked [Hakenberg1a et al., 2009] system in the competition achieved an f-measure of 35%). Therefore, the use of machine learning techniques such as SVM notably improves the performance of the PPI extraction

System	Technique	Features	Corpus	Results
[Xiao et al., 2005]	ME	lexical, syntactic and semantic	IEPA	F1=90.8%
[Chun et al., 2006]	Dictionary Matching + ME algorithm	words, PoS tags, acronyms	1,000 co-occurrences of gene-diseases	F1=82.5%
[Katrenko and Adriaans, 2007]	Dependency parsing + Naïve Bayes, BayesNet and K-nearest neighbor	lemmas, syntactic functions, parents	AIMed and LLL	F1=72.7%
[Yang et al., 2009a]	link grammar parser + SVM	surface word, keyword, protein name distance and link path features	DIP	F1=57.85%
[Angus et al., 2008]	SVM	surface word, keyword, protein name distance and link path features	77 patient narratives	F1=72%
[Chen et al., 2009]	SVM	protein names, context words, keywords, distance, number of proteins, position and distance between keyword and protein	BioCreative dataset	F1=57.8%

Table 6.3: Main feature-based machine learning approaches for biomedical relation extraction

systems.

Although feature-based methods have obtained good results in the relation extraction task, they have several drawbacks. The main shortcomings are as follows:

1. The feature extraction process requires extensive domain knowledge and much time is spent on the task of feature engineering.

2. Features are not able to correctly capture the structural information gathered in complex structures such as parse trees or dependency graphs, that is, this type of input information is not easily represented by explicit features.

3. Features from different information sources (lexical, syntactic, semantic) may be incompatibles with each other.

4. The great diversity of words produces high or even infinite dimensionality of the feature space. In these cases, the computation of the feature map becomes computationally unfeasible. In addition, the feature vectors are too sparse.

6.4.2. Kernels-based methods for Relation Extraction

Kernels-based approaches [Cristianini and Shawe-Taylor, 2000, Shawe-Taylor and Cristianini, 2004] provide an effective alternative to feature-based approaches. Their main advantage is that they can exploit the structural descriptions of words, phrases and sentences, and process them efficiently. Kernels do not need to represent each data instance into a flat set of features, but just define a similarity measure that determines the similarity between the instances. The intuitive idea behind a kernel method is to find a mapping of the input space into a new feature (vector) space in which solving the problem is easier.

Formally, a kernel function is a binary function $K : X x X-> [0, \infty)$ that maps a pair of objects $x, y \in X$ to their similarity score $K(x, y)$. The kernel function must satisfy

$$\forall x, y \in X : k(x, y) = \langle \phi(x), \phi(y) \rangle, \tag{6.1}$$

where $\phi : X \rightarrow F \subseteq \mathbb{R}^n$ is a mapping from the input space X to a vector space F. The mapping function ϕ transforms each instance $x \in X$ in a feature vector $\phi(x) = (\phi_1(x), \phi_2(x), \ldots, \phi_m(x))$, where $\phi_i : X \rightarrow \mathbb{R}$, with no need to know the explicit representation of the x. Then, the mapping function ϕ allows to express $K(x, y)$ as the dot-product of the features vectors of the input objects x and y.

$$\forall x, y \in X : k(x, y) = \langle \phi(x), \phi(y) \rangle = \sum_{i=1}^{m} \phi_i(x).\phi_i(y). \tag{6.2}$$
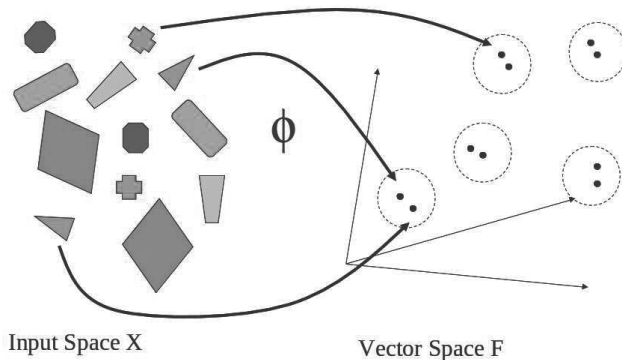
Figure 6.2: ϕ embeds the instances into a feature (vector) space. The figure is taken from [Renders, 2004]

In summary, the kernel method acts as the interface between the input data and the learning algorithm. The input objects are embedded into a vector space, in which the learning process takes place by measuring the similarity between objects [Shawe-Taylor and Cristianini, 2004]. Many machine learning algorithms can be formulated as kernel-based algorithms. The most popular ones are support-vector machines, k-nearest neighbors and voted perceptrons.

We now review the major types of kernels and approaches that have been applied to relation extraction in the biomedical domain. The performance of kernel approaches is mainly determined by the selection and design of the kernel functions, which are based on the representation of the relation instances. The instances can be presented in a certain representation such as bag-of-words, word sequences, parse trees or dependency graphs. These representations allow to capture different types of contextual information.

A bag-of-words is the simplest and most-used representation of relation instances. Each relation instance is represented as a vector where each component indicates the occurrence of a particular word in the sentence. The two entities that constitute the instance can be excluded from its word representation in order to differentiate it from the rest of the instances. Formally, each relation instance x can be represented as a vector

$$\phi(x) = (phi_1(x), phi_2(x), \ldots, phi_n(x)) \in \{0, 1\}^n \tag{6.3}$$

where $\phi_i(x)$ indicates if the word i occurs in the sentence. Therefore, a *word kernel* function (also called bag-of-words kernel) $K_{WORD}(x,y)$ can be defined on such word representation as the inner product of the vectors $\phi(x)$ and $\phi(y)$ and returns the number of words in common between two instances:

$$K_{WORD}(x, y) = \langle phi(x), phi(y) \rangle = \sum_{i=1}^{n} phi_i(x).phi_i(y) \tag{6.4}$$

The word kernel is simple and efficient. However, neither the sequence of words in sentence nor the sentence structure is captured by word kernels. The bag of words representation loses all the word order information only retaining it if the words occur or not in the instances.

A relation instance can also be represented as a sequence of words in order to take into account the sequential order of words in a sentence. The simplest way to represent an instance as a sequence is to consider all words except the two entities. For example, given the sentence *Propylthiouracil causes variations in the anticoagulant effects of drugs such as acenocoumarol, nisindione, dicumarol and warfarin*, the sequence representation for the relation instance (*propylthiouracil, acenocoumarol*) is shown in the following example:

Example 13 Word sequence representation for the relation instance (*propylthiouracil, acenocoumarol*) *causes-variations-in-the-anticoagulant-effects-of-drugs-such-as-,-nisindione-,-dicumarol-and-warfarin*

There are many other ways to represent the instances as sequences, for example, considering contiguous or non-contiguous subsequences for a given length of a string, assigning different weight to each gap, or including only particular types of features in the sequence. Based on the type of representation, it is possible to define different kernel functions to measure the similarity between two sequences counting the subsequences that they have in common. Thus, we can define various ways of counting:

1. *p-spectrum kernel* counts how many substrings of a given length (p) two strings have in common. The contiguity is necessary.

2. *Subsequence kernel* in which the contiguity is not necessary.

3. *Gap-weighted subsequence* kernel in which the contiguity is penalized.

The p-spectrum kernel is the most natural way of comparing two strings. This kernel function was proposed by Lodhi et al. [2002] for text classification. Let us give a simple example to explain the construction of a 2-spectrum kernel. Given the strings *'bar','bat','car' and 'cat'*, their 2-spectra ϕ is shown in table 6.4. All the other dimensions indexed by other strings of length 2 have value 0, and for this reason, they have not been shown. The resulting kernel matrix is shown in table 6.5. A detailed description of the main sequence kernel functions can be found in [Shawe-Taylor and Cristianini, 2004]. We now review the main approaches based on sequence kernel methods.

ϕ	ar	at	ba	ca
bar	1	0	1	0
bat	0	1	1	0
car	1	0	0	1
cat	0	1	0	1

Table 6.4: Example of ϕ_2 taken from [Shawe-Taylor and Cristianini, 2004]

K	bar	bat	car	cat
bar	2	1	1	0
bat	1	2	0	1
car	1	0	2	1
cat	0	1	1	2

Table 6.5: Example of 2-spectrum kernel taken from [Shawe-Taylor and Cristianini, 2004]

Bunescu and Mooney [2006] propose a generalization of the sequence kernel given in Lodhi et al. [2002]. The new kernel uses sequences that combine words and word classes. Several experiments are performed for extracting PPIs from biomedical corpora and top-level relations from newspaper corpora. They observed that if a sentence contains a relationship between two entities, then the relation is generally represented by some of the following three patterns:

1. *Fore-Between*: words before and between the two entities are used to express the relationship. For example, *interaction of Aspirin with Heparin*

2. *Between*: only words between the two entities are essential for asserting the relationship. For example, *Aspirin interacts with Heparin*.

3. *Between-After*: words between and after the two entities are used to express the relationship. For example, *Aspirin and Heparin interact*.

Then, given two sentences *s* and *t*, the kernel relation computes the number of common patterns between *s* and *t*, only considering the above set of patterns. Therefore, the kernel *K(s,t)* is expressed as the sum of the three sub−kernels above, as follows:

$$K(s,t) = K_{before}(s,t) + K_{between}(s,t) + K_{after}(s,t) \qquad (6.5)$$

They also noted that all these patterns use at most 4 words to express the relationship, not counting the two entities. Thus, they only considered the subsequences that satisfy some of the three patterns above, with a maximum word−length of 4, for defining the subsequence kernel. The patterns are completely lexicalized and consequently their performance is limited by data sparsity. In order to alleviate the problem of the sparsity, the sequences are represented using words and word classes such as PoS tags and entity types. The sequence kernel was evaluated on the AImed corpus, achieving a precision of 65%, a recall of 46.4% and an f-measure of 54.2%. Experiments were also performed on the ACE 2002 corpus, which allowed to compare this approach with other types of kernels applied to relation extraction in the general domain, showing that the sequence kernel achieves a better performance than the tree kernel introduced by Culotta and Sorensen [2004].

Based on the work above, Giuliano et al. [2006] propose two kernels to represent two distinct information sources: (1) the *global context* where entities appear and (2) the *local context* of each entity. Thus, the whole sentence where the entities appear (global context) is used to discover the presence of a relation between two entities. Windows of limited size around the entities (local contexts) provide useful clues to identify the roles played by the entities within a relation (for example, agent and target of a gene interaction). Then, they defined a third kernel as the sum of the global context and the local context kernels. This kernel is called *shallow linguistic kernel* because it is based solely on shallow linguistic processing, such as tokenization, sentence splitting, part of speech (PoS) tagging and lemmatization. Experiments were performed on the two biomedical corpora AImed and LLL. Table 6.6 shows the results obtained by this work adopting the evaluation methodology *OAOD* (one answer per occurrence in the document) [Lavelli et al., 2004], that is, each individual occurrence of a protein interaction must be extracted from the document. In this thesis, we have applied this shallow linguistic kernel to the extraction of DDIs. A more detailed description of this kernel can be found in section 8.2.

Corpora	Kernel	P	R	F1
AImed	$K_{GlobalContext}$	57.7%	60.1%	58.9%
	$K_{LocalContext}$	37.3%	56.3%	44.9%
	$K_{ShallowLinguistic}$	60.9%	57.2%	59.0%
LLL	$K_{GlobalContext}$	55.1%	66.3%	60.2%
	$K_{LocalContext}$	44.8%	60.1%	53.8%
	$K_{ShallowLinguistic}$	62.1%	61.3%	61.7%

Table 6.6: Results of the work [Giuliano et al.,2006] for PPI extraction

A parse tree is a clear example of structure data representing the syntactic structure of a sentence (see figure 6.3). This kind of structural information can be very beneficial to detect relations between entities. Thus, the relation instances can be represented as parse trees, and a tree kernel function measures the similarity between two relation instances. As it was aforementioned, an important property of the kernel methods is their ability to retain the structural information of relation instances. We can define a *tree kernel* as a function $K_{TREE}(T_1, T_2)$ that returns a similarity score for the two trees T_1 and T_2. The tree kernels can be considered as *convolution kernels* since they can calculate the similarity between two trees in a recursive manner by estimating the similarity of their subtrees [Collins and Duffy, 2002]. Consequently, the tree kernels are able to calculate the structural similarity effectively.

Below, the main approaches based on tree kernels for relation extraction are reviewed. We begin by reviewing the main works in the newswire domain, which have been the basis for the approaches developed in the biomedical domain.

Zelenko et al. [2003] were the first using tree kernels for the relation extraction task. Sentences are represented as shallow parse trees. A relation instance is represented as the smallest shallow subtree that contains both entities. Each node in the tree is augmented with several attributes such as its head (if it is a phrase), its PoS tag (noun, verb, etc), its entity type (for example, person, organization, etc), or its role in the relation. These attributes are used to estimate the similarity between the nodes. Previously to define the kernel function on shallow parse trees, they define a primitive kernel function to estimate the similarity between the nodes. This similarity function on nodes is computed in terms of the attributes of the nodes. Then, they define the kernel method in terms of similarity function of their root nodes and the similarity of their children. The kernels were applied to extract two types of relations (person-affiliation,
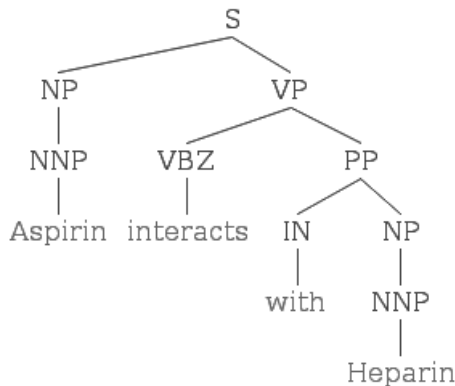
Figure 6.3: The smallest subtree containing both drugs is the whole parse tree

organization-location) and evaluated on a corpus of 200 news articles. The F-measure of the results ranged from 80% to 86%.

[Culotta and Sorensen, 2004] propose an extension of the previous approach by a richer sentence representation and the use of composite kernels to reduce kernel sparsity. The main hypothesis is that instances containing similar relations will share similar substructures in their dependency trees. The sentences are analyzed by the MXPOST parser [Ratnaparkhi, 1996] in order to generate their parser trees, which are transformed to dependency trees by a set of rules (for example, subjects are dependent on their verbs and adjectives are dependent on the nouns they modify). Based on the work [Zelenko et al., 2003], they represent each relation instance as the smallest common subtree in the dependency tree that includes both entities. This choice allows to reduce noise and emphasize the local characteristics of relations. In addition, each node is augmented with a feature vector that includes attributes such as the word, its part of speech tag, its entity type, its WordNet [Fellbaum et al., 1998] hypernyms, and a relation argument indicating if an entity is the first or second argument in a relation. Then, a tree kernel is defined on dependency trees to estimate the similarity between relation instances. They also proposed a bag-of-words kernel, which treats the tree as a vector of features, that was also developed and combined with the tree kernels. Experiments were performed on the ACE corpus, showing a precision of 67.1%, a recall of 35.0%, and f-measure of 45.8%. Another interesting result was that the tree kernel outperformed the bag-of-words kernel, implying that the structural information is extremely useful for relation detection. The approach presented in [Bunescu and Mooney, 2006] seems to obtain better results, however, these are not directly comparable since the approaches used different corpora for performing their experiments.

Based also on the use of the dependency trees, Bunescu and Mooney [2005] designed a kernel method that uses the shortest path between the two entities in a dependency tree, instead of the smallest subtree containing both entities proposed in the previous works [Zelenko et al., 2003, Culotta and Sorensen, 2004]. Because the dependency graph is always connected, it is guaranteed to find a shortest path between the two entities. Thus, each relation instance is represented as the shortest path between the two entities. As these paths are completely lexicalized and consequently their performance will be limited by data sparsity, they decided to add classes such as the PoS tags or entity types to the representation of the path. They defined a simple kernel whose basic idea is that if the two paths have different lengths, they correspond to different ways of expressing a relationship, and otherwise, the similarity is the product of the number of common word classes at each position in the two paths. Formally, given two relation examples $x=x_1 x_2 \dots x_m$ and $y=y_1 y_2 \dots y_m$, where x_i denotes the set of word attributes corresponding to position i, the kernel method is defined as follows

$$k(x,y) = \begin{cases} 1, & n \neq m \\ \prod_{i=1}^{n} c(x_i, y_i), & m = n \end{cases} \tag{6.6}$$

where $c(x_i, y_i) = |x_i \cap y_i|$ is the number of common classes between x_i and y_i. The kernel was used in conjunction with SVMs in order to find the hyperplanes that best separate positive examples from

negative examples. ACE corpus 2002 was used in the experiments. The documents were processed using the OpenNLP package [Baldridge, 2004]. In addition, they used the entities provided by the gold-standard corpus. It yields better results than if the entities are recognized automatically. The shortest-based kernel achieves a precision of 65.5% and recall of 43.8%. Results showed that this kernel achieved better performance (f-measure 52.5%) than the composite kernel (f-measure=45.8%) proposed in [Culotta and Sorensen, 2004].

The approach presented in [Eom et al., 2006] used a tree kernel-based method for PPIs extraction. They retrieved 2000 abstracts by querying with keywords such as *protein interaction* and the name of concrete proteins. The abstracts are segmented into sentences. Only those sentences with at least two proteins and interaction-related words are selected. The sentences that include PPIs were manually labeled as positive, otherwise labeled as negative. The final corpus consists of 1135 sentences of positive examples, and 569 sentences of negative ones. The sentences are analyzed by the parsers Brill tagger [Brill, 1992] and Collins parser [Collins, 1996]. Based on [Zelenko et al., 2003], they only considered the smallest subtrees that contained the two proteins. To improve the performance, the subtree is augmented with semantic information such as entity type or interaction-related words. Nodes that are proteins are labeled with the tag *PTN* (protein), and nodes that are interaction-related words are also modified in order to differentiate from general words. They performed 10-fold cross validation to evaluate the system. Experiments achieved an f-measure of 90.48%, however, the performance cannot be compared to the results presented in [Giuliano et al., 2006] or [Bunescu and Mooney, 2006], because Eom et al. [2006] do not use any gold standard corpora such as AIMed or LLL. Experiments showed that the approach achieves better performance than a bag-of-word kernel or a naïve bayes method.

Li et al. [2008] compared and combined different kernels for biomedical relation extraction, in particular, a bag-of-words kernel, a subsequence kernel and the tree kernel proposed by [Zelenko et al., 2003], which is augmented by incorporating from the root node of the smallest subtree to the root of the full parse tree. Thus, this trace is a sequence word that captures more global context in the full parse tree in addition to the smallest subtree. The sequence kernel is used to compare the similarity between traces. In order to evaluate the kernels, they built a corpus of 2,000 cancer-related abstracts from Medline. The biomedical entities and relations between them were manually annotated by a biomedical expert. The sentences were analysed by the biomedical parser presented in [Lease and Charniak, 2005]. The corpus contains a total of 8,071 relation instances, 2,156 of them were identified as true relations, while the remaining were labeled as negative examples. The best results were achieved by the composite kernel that combines the sequence kernel and the trace-tree kernel. This composite kernel achieved an f-measure of 67.23% (R=64.68%, P=70.11%, Accuracy=83.14%). In addition, the experiment showed that the tree kernel significantly outperforms the sequence and word kernels. This is due to the fact that the tree kernel is able to better capture the structural information between the entities. In addition, sequence and word kernels may include some noise since they require comparing all the words and sequences in sentences. The augmented tree kernel also improved the results with respect to the standard tree kernel.

Airola et al. [2008] proposed a dependency-path kernel to extract PPIs. Each relation instance is represented with a weighted graph that consists of two unconnected subgraphs. The former represents the dependency structure of the sentence, and the second the linear order of the words. Based on the *shortest path hypothesis* [Bunescu and Mooney, 2005], the nodes on the shortest paths connecting the two proteins are differentiated with a special tag and all the edges on the shortest path receive a higher weight, emphasizing the shortest path without disregarding information outside the path. The second subgraph represents the linear structure of the sentence. In this graph, information such as text, POS tags, entity types and the order with respect to the proteins is denoted. Experiments were performed across five corpora annotated for PPIs: Aimed, BioInfer, HPRD50, IEAP and LLL (see chapter 3), which were processed with the Charniak-Learse parse [Lease and Charniak, 2005], and the parse tree were transformed into the dependency graphs by the Standford tools. They evaluated the kernel with 10-fold document-level cross-validation on all of the corpora. The experiments showed that the values of f-measure vary remarkably between the different corpora, with much higher results on IEPA and LLL than on AImed and BioInfer.

Since f-measure is very sensitive to the underlying positive/negative pair distribution of the corpus, they also calculated the AUC measure. However, the AUC measure is invariant and falls in range of 83-85%. This is due to the fact that the f-measure is not invariant to the distribution of positive and negative examples. Thus, the greater the fraction of true interactions in a corpus is, the easier it is to reach high performance in terms of f-measure. The fraction of true interactions out of all candidates is 50% on the LLL corpus, but only 17% on AImed. The best f-measure (56.4%) was achieved on the AImed corpus.

To the best of our knowledge, the only approach dealing with a ternary relation extraction task is proposed in [Liu et al., 2007]. PROTEIN-ORGANISM-LOCATION relations are identified in the text of biomedical articles. Different kernel functions are used with an SVM learner to integrate two sources of information from syntactic parse trees: (1) a large number of syntactic features useful for semantic role labeling task, and (2) features from the entire parse tree using a tree kernel. The best result is obtained by combining semantic role labeling features with tree kernels (P=75.3%, R=74.5%, F=74.9%). The experiments showed that the use of rich syntactic features significantly outperforms shallow word-based features.

Although many approaches have been addressed for the biomedical relation extraction task, its direct comparison is impossible since there are many differences not only in the set of solved relationships, but also the evaluation resources. If we compare feature-based and kernels-based approaches evaluated on the AIMed, we observe that kernels do not achieve to overcome the feature-based approaches. For example, the feature-based presented in [Katrenko and Adriaans, 2007] achieved an f-measure of 72% compared to 54.2% obtained by Bunescu and Mooney [2006] or to 59.0% obtained by Giuliano et al. [2006]. The tree kernel-based approach presented in [Eom et al., 2006] seems to obtain better performance (f-measure=90.48%) than these sequence kernels, however, they are not directly comparable since Eom et al. [2006] used a private dataset for performing their experiments. Li et al. [2008] proposed an interesting comparasion of several kernels. Experiments showed that the tree kernels overcame the other kernels. The best performance was achieved by the combination of the different kernels. Unfortunately, a private dataset was used for the evaluation, and therefore, this approach cannot be compared directly with others. Airola et al. [2008] evaluated a dependency-path kernel on five different corpora for PPIs extraction. The experiments showed that the best results (56.4%) are achieved on the AImed corpus. Therefore, the sequence kernel proposed by Bunescu and Mooney [2006] seems to be so far the most successful method for PPIs extraction on the corpus AIMed. Some approaches [Zelenko et al., 2003, Li et al., 2008] have declared that tree kernels not only outperform feature-based methods, they also achieve better results than sequence kernels. However, we believe that it is not possible to draw real conclusions on the different approaches, until the experiments are performed on the same corpus.

On the other hand, tree kernels are relatively slow compared to feature classifiers and sequence kernels [Bunescu and Mooney, 2005, Li et al., 2008]. The complexity of tree kernels and the need to evaluate thousands of them during the process of classification may render them inappropriate for practical purposes. For this reason, we believe that sequence kernels are more appropriate than tree kernels for DDIs extraction since these methods should be integrated into a real application in which the processing time will be a priority.

System	Kernel	Corpora	Results
[Bunescu and Mooney, 2006]	sequence kernel	AIMed	F1=54.2%
[Giuliano et al., 2006]	sequence kernel	AIMed, LLL	F1=61.7%
[Eom et al., 2006]	tree kernel	2000 abstracts	F1=90.48%
[Li et al., 2008]	bag-of-words, subsequence, tree, trace kernels	2000 abstracts	F1=67.23%
[Airola et al., 2008]	dependency-path kernel	Aimed, BioInfer, HPRD50, IEAP and LLL	F1=56.4% (AIMed)

Table 6.7: Main kernel-based machine learning approaches for biomedical relation extraction

6.5. Unsolved Issues in Biomedical Relation Extraction

Relation Extraction in biomedicine has been studied for approximately ten years. Over these years, biomedical relation extraction systems have grown from simple rule-based pattern matcher to sophisticated, hybrid parser employing computational linguistics technology or machine learning methods. But, until now, there are still several severe obstacles to overcome.

With regard to the type of relations addressed, most works have focused on the extraction of biological relationships such as genetic and protein interactions. To the best of our knowledge, there is no approach that tackles the automatic detection of DDIs from biomedical texts. In fact, the only approaches [Duda et al., 2005, Guo and Ramakrishnan, 2009] focused on DDIs deal with the classification of articles about

DDIs, but do not extract them explicitly. Several approaches [Kolarik et al., 2007, Garten and Altman, 2009] have addressed the extraction of relevant information for the pharmacological domain such as drug properties, pharmacogenomic concepts, gene-drug interactions, among others, however, none of them has dealt with the extraction of relevant information for DDI such as the mechanism, the relation to the drug dosages, the time course, the factors that alter an individual's susceptibility, seriousness and severity, the probability of the occurrence, etc.

Regarding the performance of the existing works to cope with the relation extraction problem, firstly we should note that though a wide range of methods have been addressed, its direct comparison is impossible since there are many differences not only in the set of solved relationships, but also the evaluation resources and strategies. At this point, we must highlight the crucial role of the BioCreative challenges in improving the text mining techniques in the biological domain, providing a common framework for evaluation. A general characteristic, common to almost all existing works, is the use of information from different levels of linguistic analysis. Thus, the results depend heavily on good results from previous tasks such as text processing tasks (tokenization, sentence splitting, PoS tagging, syntactic parsing) or named entity recognition. Therefore, until these tasks do not deliver good results, the relation extraction task will continue to deliver relatively poor results.

Linguistic-based approaches and pattern-based approaches perform well for capturing relatively simple binary relationships between entities in a sentence, but fail to extract more complex relationships expressed in various clauses. The results reported by these methods are very disparate. In linguistic-based approaches, f-measure ranges from about 39% to 80%. The worst results are reported by the IntEx system based on the use of link grammar parser, while the best ones are achieved by the RelEx system based on dependency parsing. Both approaches were evaluated on different corpora, so it is daring to draw any conclusion about whether the good performance of the RelEx system is due to the dependecy parsing or to the corpus. However, we should note that the BioPPIExtractor, evaluated on the DIP corpus, achieves lower results than the RelEx system, though both systems are based on dependency parsing. Regarding the performance of the pattern-based methods, their f-measure range from 43% to 71% (but Ono et al. [2001] reported an f-measure of 90.2%). Linguistic-based approaches and pattern-based approaches are not able to handle the complexity of the biomedical sentences. Most works do not deal with adverbial and prepositional phrases, which are essential to describe biomedical relations. In biomedical texts, many relationships are often described through disjoint clauses, however, this issue has been rarely addressed. Moreover, very few approaches [Sánchez-Graillet and Poesio, 2007, Morante and Daelemans, 2009] take into account other important aspects of sentence constructions such as mood, modality and negation which can significantly alter or even reverse the meaning of the sentence.

In general, machine learning-based approaches have achieved better performance than linguistic-based and pattern-based ones, as demonstrated in the last BioCreative challenge. However, machine learning-based approaches depend heavily on the annotated corpora for training and testing. Most available corpora in biomedical domain focus on PPIs. Corpus annotation is an expensive work, usually involving an extensive time and labor. Regarding feature-based methods, they usually use lexical, syntactic and semantic features. However, they are not able to correctly capture the structural information gathered in parse trees or dependency graphs. The structural information is extremely useful for relation detection. Kernel-based methods do not need to represent each data instance into a flat set of features, but just define a similarity measure that determines the similarity between the instances. In the kernel methods, the instances can be presented in a certain representation such as bag-of-words, word sequences, parse trees or dependency graphs. As it was aforementioned, the comparison among the different works is impossible since they have evaluated their experiments on different corpora. Among the works evaluated on the same corpus, the sequence kernel presented in [Bunescu and Mooney, 2006] achieves the best performance. However, this approach does not achieve to overcome the results obtained by the feature-based method presented in [Katrenko and Adriaans, 2007]. Other works [Eom et al., 2006, Li et al., 2008] declare that the tree kernels achieve better performance, however, they were evaluated on private datasets. Therefore, it is essential to make an evaluation of the different kernels using a common corpus in order to draw real conclusions.

Biomedical relation extraction methods generate poorer results compared with other domains such as newswire. A possible cause is that few approaches take advantages of the information about relations included in ontologies [Huang et al., 2004b, 2006b], which usually are used like simple dictionaries. We believe that a more extensive and effective use of the biomedical ontologies is a prerequisite to improve the performance of the biomedical relation extraction task. Also, biomedical text needs to be semantically annotated and actively linked to ontologies. Although, in general, relationships can involve three o more entities, most of the existing approaches for biomedical relation extraction have focused on the extraction

of binary relationships. Moreover, the existing approaches usually assume that the argument entities of the relation occur in the same sentence. Consequently, the extraction of the complex relationships as well as the relations spanning several sentences are some of the remaining challenges.

The continuing growth and diversification of the scientific literature will require tremendous systematic and automated efforts to utilize the underlying information. Therefore, in the near future, tools for knowledge discovery will play a pivotal role in biomedical systems. The increasing fervor in the field of biomedical IE gives the evidence [Zhou and He, 2007]. However, we should point out that there is a huge gap between life science researches, healthcare professionals and computational scientists. Bridging the gap between life science scientists and computational scientists is crucial to the success of biomedical IE. Currently, this field is dominated by researchers with computational background; however, the biomedical knowledge is only possessed by life science scientists. That is crucial for defining standards for evaluation; for identification of specific requirements, potential applications and integrated information system for querying, visualization and analysis of data on a large scale; for experimental verification to facilitate the understanding of biological interactions. Hence, to attract more biologists and healthcare professionals into the field, it is important to design simple and friendly user interfaces that make the tools accessible to non-specialists.

Chapter 7

Combining syntactic information and patterns for Drug-Drug Interaction extraction

7.1. Introduction

This thesis proposes two different approaches for DDIs extraction task: (1) a hybrid approach that combines syntactic information and pharmacological patterns, and (2) a kernel method for DDI extraction. This chapter describes in detail the first approach, which exploits the information provided by the previous processes (see figure 7.1) text analysis, drug name recognition and anaphora resolution.

Although many approaches have been proposed to extract biomedical relations, only a few of them achieve successful results. One important reason is that only a few approaches [Huang et al., 2006a] have dealt with the issue of the grammatical complexity of biomedical texts. However, language structures such as apposition, coordination and complex sentences are very common in the biomedical literature. We think that the detection of these linguistic phenomena is essential to successfully tackle the extraction of biomedical relations, in particular, DDIs.

In this chapter, we propose a hybrid method, which combines shallow parsing and pattern matching, to extract relations between drugs from biomedical texts. A pharmacist has defined a set of lexical patterns to capture the most common expressions of DDIs in texts, based on her professional experience and the corpus observation. The method is based on the approach described in [Huang et al., 2006a], which proposes a set of syntactic patterns to split the long sentences into clauses from which relations are extracted by a pattern matching algorithm. This approach works on the detection of appositions, coordinate constructions and relative clauses. Our contribution extends this approach detecting any kind of subordinate and coordinate clause. Appositions and coordinate structures are interpreted based on shallow syntactic parsing provided by MMTx. In particular, we define a set of syntactic patterns to detect them. Subsequently, complex and compound sentences are broken down into clauses from which simple sentences are generated by a set of simplification rules. Finally, the lexical patterns are matched with the generated sentences in order to extract DDIs.

At this point, we raise the following question: what depth of analysis is required to identify and resolve the aforesaid mentioned syntactic structures?. The complexity of the biomedical language presents a compelling computational challenge for any parser. Complex and long sentences tend to cause problems due to their long length and high degree of ambiguity [Siddharthan, 2006]. On the other hand, full parsers are less robust and computationally more expensive than shallow parsers. Moreover, to the best of our knowledge, Genia Dependency parser [Sagae, 2007] and the parser presented in [Lease and Charniak, 2005] are the only full parsers trained in biomedical texts. In our case, shallow parsing is feasible because we only need to identify a limited range of grammatical functions and clauses, and full grammatical relations are not needed. Therefore, we have decided to use the information provided by MMTx for detecting appositions, coordinate structures and clauses.

The chapter is organized as follows. The treatment of coordinate structures and appositions are described in sections 7.2 and 7.3 respectively. Section 7.4 shows how clauses boundaries are identified using shallow syntactic information and how simple sentences are generated from the clauses. Section 7.5 presents a shallow evaluation of the performance in the resolution of syntactic structures tackled in this
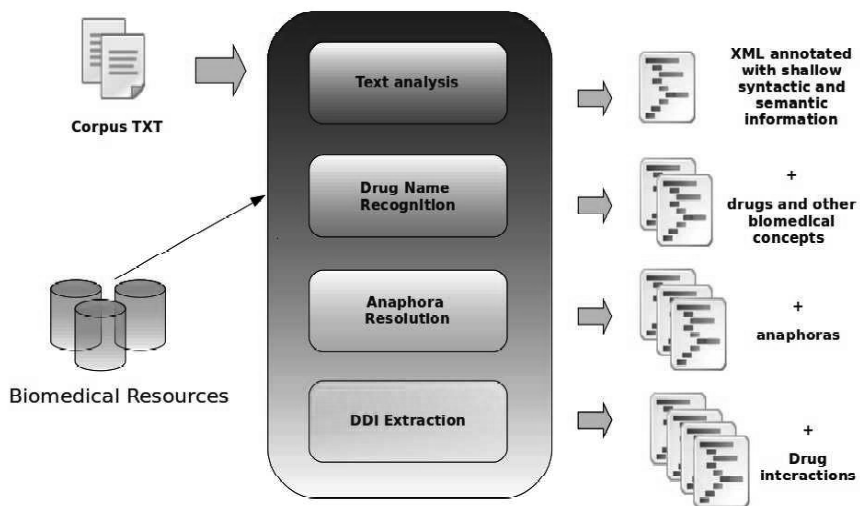
Figure 7.1: DDI Extraction Prototype.
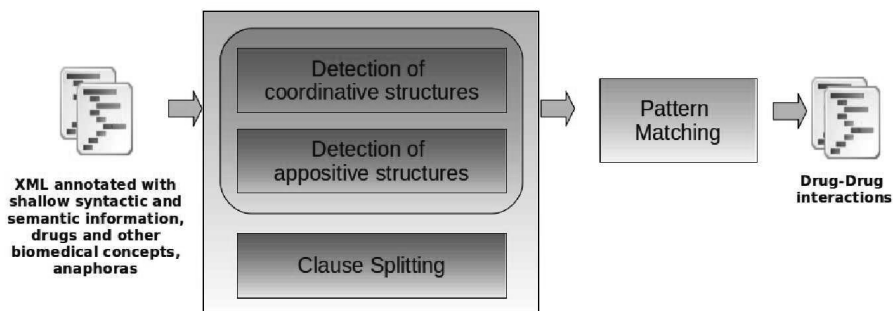
Figure 7.2: Outline of the pattern-based approach to extract DDIs.

chapter. Section 7.6 introduces the set of pharmacological patterns proposed by our pharmacist. Section 7.7 describes in detail the experiments and presents the experimental results. Finally, conclusions and future work are presented in section 7.8.

7.2. Detecting coordinate structures

Coordination in natural language occurs in a wide range of constructions. The diversity of sentence types in which coordination occurs has led to it being one of the most hotly debated, and yet relatively

little understood, issues in Linguistic [Dik, 1968, Fabricius-Hansen and Ramm, 1984, Johannessen, 1998, Quirk et al., 1985]. A coordination is a complex syntactic structure that joins two or more sentence constituents, called conjuncts or conjoines [Van Oirsouw, 1987]. For example, in the following coordinate structure *probenecid, sulfinpyrazone, and phenylbutazone*, the conjunction *and* coordinates the conjunct *probenecid* with *sulfinpyrazone* and with *phenylbutazone*. Coordination is an extremely common grammatical phenomenon in biomedical texts. Coordinate structures can be illustrated by the italic part of the following sentences taken from DrugBank:

Example 14 Sentences containing coordinate structures.

(1) *Broad-spectrum antibiotics may sterilize the bowel and decrease the vitamin K contribution to the body by the intestinal microflora.*

(2) Population pharmacokinetic analyses revealed that *NSAIDs, corticosteroids, and TNF blocking agents* did not influence abatacept clearance.

(3) These agents *may bind and decrease* absorption of levothyroxine sodium from the gastrointestinal tract.

(4) Proscar is *toxic and dangerous*.

Sentence (1) shows conjoined sentences, but coordination is also possible for noun phrases as in sentence (2), verb phrases as in the sentence (3), and adjectival phrases as in sentence (4). The conjoined sentences are addressed in section 7.4, since their treatment is directly related to the division of clauses. The work deals with the other types of constituents.

Figure 7.3 shows a sentence that contains three interactions. In order to extract the interactions, it is necessary to interpret the coordinate structure in it. Since coordinate constituents are semantically close and usually they play the same syntactic and grammatical roles in a sentence, it is necessary to assemble them together [Huang et al., 2006a].

Figure 7.3: Example of coordinate structure.

Although a wide variety of structures can be conjoined, not all coordinations are acceptable. Chomsky [2002] concluded that syntactically different categories cannot be conjoined. This constraint is known as *Coordination of Likes Constraint (CLC)* or *Law of Coordination of Likes* [Williams, 1978]. Coordinating conjunctions, also called coordinators, are conjunctions that connect two or more items of equal syntactic importance (words, phrases or clauses). Coordinating conjunctions include *for, and, nor, but, or, yet,* and *so*. However, since this section focuses on coordination between phrases, we have only considered coordinators shown in table 7.1 as possible coordinators to link phrases. The rest (*for, but, yet*, and *so*) are conjunctions that connect clauses.

Coordinators:	AND | OR | NOR | AND/OR | AS WELL AS

Table 7.1: Coordinators to link phrases.

Based on the *CLC* constraint, we initially proposed the set of syntactic patterns shown in table 7.2 to detect coordinate structures. However, based on the corpus observation, this constraint is too restrictive for the kind of parsing provided by MMTx. For example, the sentence shown in figure 7.4 demonstrates that being of the same syntactic category is too strong requirement for conjuncts in a coordinate construction, since a prepositional phrase, *of probenecid (s0.p6)*, can be conjoined with two noun phrases: *sulfinpyrazone (s0.p7)* and *phenylbutazone (s0.p9)*.

Pattern	Example
(NP,)* NP CONJ NP	[NSAIDs]$_{NP}$, [corticosteroids]$_{NP}$, and , [TNF blocking agents]$_{NP}$ did not influence abatacept clearance.
(ADJ,)* ADJ CONJ ADJ	Proscar is [toxic]$_{ADJ}$ and [dangerous]$_{ADJ}$
(VP,)* VP CONJ VP	These agents may [bind]$_{VP}$ and [decrease]$_{VP}$ absorption of levothyroxine sodium from the gastrointestinal tract.

Table 7.2: Patterns to detect coordinate structures.

```
−<SENTENCE ID="s0" TEXT="Uricosuric Agents: Aspirin may decrease the effects of
  probenecid, sulfinpyrazone, and phenylbutazone.">
  −<PHRASES>
    +<PHRASE ID="s0.p0" NUMTOKENS="2" TEXT="Uricosuric Agents" TYPE="NP">
      </PHRASE>
    +<PHRASE ID="s0.p1" NUMTOKENS="1" TEXT="" TYPE="UNK" USAN="NO">
      </PHRASE>
    +<PHRASE ID="s0.p2" NUMTOKENS="1" TEXT="Aspirin" TYPE="NP"></PHRASE>
    +<PHRASE ID="s0.p3" NUMTOKENS="1" TEXT="may" TYPE="VP"></PHRASE>
    +<PHRASE ID="s0.p4" NUMTOKENS="1" TEXT="decrease" TYPE="VP"></PHRASE>
    +<PHRASE ID="s0.p5" NUMTOKENS="2" TEXT="the effects" TYPE="NP"></PHRASE
    +<PHRASE ID="s0.p6" NUMTOKENS="3" TEXT="of probenecid" TYPE="PP/of">
      </PHRASE>
    +<PHRASE ID="s0.p7" NUMTOKENS="2" TEXT="sulfinpyrazone" TYPE="NP">
      </PHRASE>
    +<PHRASE ID="s0.p8" NUMTOKENS="1" TEXT="and" TYPE="CONJ"></PHRASE>
    +<PHRASE ID="s0.p9" NUMTOKENS="2" TEXT="phenylbutazone" TYPE="NP">
      </PHRASE>
  </PHRASES>
  +<DDIS></DDIS>
</SENTENCE>
```

Figure 7.4: Parsed sentence by MMTx

We have observed in the training corpus that coordinate structures involving constituents with different syntactic categories are very common. Sometimes it is due to the fact that MMTx is not able to determine the syntactic type of a phrase, classifying it as an unknown phrase (that is, with the tag *UNK*) (see example 15). Therefore, the above syntactic patterns have been merged to contemplate the parsing peculiarities of MMTx. Table 7.3 presents the new patterns where the first row shows a pattern in which different syntactic types can be combined to detect coordination at the phrase level. An exception is made for verb phrases, since the coordination between a verbal phrase and another type of syntactic phrase is a coordination between clauses (which is tackled in the following section). Thus, the second pattern only allows to connect the verbal phrases with verbal phrases. The table 7.3 also includes a syntactic pattern, which was already introduced for drug anaphora resolution in chapter 5, to detect correlative expressions such as *both midazolam and triazolam* (see section 5.4).

Extended patterns for coordinate structures
([NP|PP|ADJ|UNK],)* [NP|PP|ADJ|UNK] CONJ [NP|PP|ADJ|UNK]
(VP,)* VP CONJ VP
[BOTH|EITHER|NEITHER][NP|PP|UNK] [AND|OR|NOR] [NP|PP|UNK]

Table 7.3: Extended patterns to detect coordinate and correlative structures.

In order to match the syntactic patterns on a sentence to detect its coordinate structures, the sequence of the syntactic types of its phrases must be generated from the shallow syntactic information provided by MMTx. If some pattern matches the sequence, recognizing one or more structures, then the text of the sentence is regenerated from the text of its phrases, except the interpreted structures which are encapsulated and replaced by the tag *COORD*. Finally, the lexical patterns defined by the pharmacist are matched with the generated text. If an interaction is detected, the coordinate structures involved

Example 15 Coordinate structures containing *UNK* phrases.

(1) The data do not suggest the need for dose adjustment of either *[Humira]$_{UNK}$* or *[MTX]$_{NP}$*

(2) When *[Inspra]$_{UNK}$* and *[NSAIDS]$_{NP}$* are used concomitantly, patients should be observed to determine whether the desired effect on blood pressure is obtained.

in it must be unfolded in order to obtain the individual interacting elements. This procedure will be described in more detail in section 7.7.

7.3. Identifying appositions

There are divergent views within Linguistics with regard to what is or is not an apposition (also called appositional or appositive structure). Fries [1952] and Francis [1958] restrict the category of apposition to coreferential noun phrases (called appositives) that are juxtaposed and refer to the same extralinguistic entity. Curme [1963] and Jespersen and McCawley [1984] expand this definition with the inclusion of constructions such as clauses and sentences as possible elements of an apposition. Burton-Roberts [1975] admits as apposition only those constructions which can be linked by a marker of apposition (see table 7.5). Quirk et al. [1985] propose three conditions to define the apposition phenomena: (1) each of the appositives can be separately omitted without affecting the acceptability of the sentence, (2) each appositive has the same syntactic role in the resultant sentences, and (3) there is no difference between the original sentence and either of the resultant sentences in extra-linguistic reference. Quirk et al. [1985] also classify apposition into various semantic classes such as exemplification, appellation, identification, and particularization.

Although the above approaches provide insights into the category of apposition, they provide either an inadequate or an incomplete description of apposition. The objective of this thesis is not to provide formal and complete description of apposition, but rather to identify appositions, in particular, those that contain drugs. We only deal with appositions that are linked by a marker of apposition. This kind of apposition appears frequently in the sentences that contain DDIs. Markers are helpful clues for detecting these structures. Table 7.5 shows the markers of apposition that we have used in this approach. Other markers such as *particularly*, *that is* or *especially* have not been tackled. Appositions that are not linked by any marker are also frequent in scientific texts, however, the lack of marker makes the detection of this kind of apposition extremely difficult. Moreover, we have observed they hardly ever occur in expressions describing DDIs.

Patterns	
APPOSITIVE	[*NP*|*PP*|*UNK*|*APPOSITION*]
APPOSITION	*APPOSITIVE*(,)? (()? *MARKER* [*APPOSITIVE*(,)?]+ (*AND*|*OR*)? (*APPOSITIVE*)? ())?

Table 7.4: Patterns to detect appositions.

We define a set of syntactic patterns in order to identify the appositions (see table 7.3). Appositions comprise at least two contiguous phrases, the second of which is marked by clues such as parentheses or markers (table 7.5). This second phrase may be a coordinate structure. The first pattern *APPOSITIVE* allows to recognize the intervening elements in an apposition, that is, their appositives. This pattern matches a phrase type (provided by MMTx) or another apposition. In this way, the pattern is able to recognize nested appositions. Regarding the phrase types, it has not considered types such as *VP*, *CONJ*, *ADV*, or, *ADJ*, since our main focus is to recognize appositions containing drugs. Drugs only appear in noun, preposition and unknown phrases. The second pattern is used to recognize appositions. This pattern matches an intervening element *APPOSITIVE* followed by a marker and by one or more intervening elements expressed by coordinate phrases. Parentheses are also included in the pattern.

Example 16 presents some sentences in which its appositions (italic part) have been recognized by the above patterns. The last sentence contains the apposition *Catecholamine-depleting drugs, such as reserpine*, which satisfies the conditions defined in [Quirk et al., 1985]. That is, both appositives share the same syntactic role in the sentence, and if they are separately omitted from the original sentence, both resultant sentences are acceptable. Two different DDIs (*Catecholamine-depleting drugs* with *beta-blocking*

Marker	Example
such as	*Catecholamine-depleting drugs, such as reserpine,* may have an additive effect when given with beta-blocking agents.
like	Buspirone does not displace tightly bound *drugs like phenytoin, propranolol, and warfarin* from serum proteins.
including	Concomitant use of apomorphine with *drugs of the 5HT3 antagonist class (including, ondansetron, granisetron, dolasetron, palonosetron, and alosetron)* is contraindicated.
for example	Propylthiouracil may increase the effect of *oral blood thinners, for example warfarin.*
e.g.	Interactions could occur following concomitant administration of *psychotropic drugs (e.g., narcotics, analgesics, antiemetics, sedatives, tranquilizers).*
i.e.	Diethylpropion may interfere with *antihypertensive drugs (i.e., guanethidine, a-methyldopa).*

Table 7.5: Markers of apposition.

agents, and *Reserpine* with *beta-blocking agents*) can be extracted from the sentence. Therefore, it is essential to detect and resolve the appositions occurring in sentences, prior to the application of the lexical patterns responsible for DDI extraction. The appositions are firstly encapsulated and then unfolded when the relation is obtained by any lexical pattern. Section 7.7 describes in detail the stage of matching.

Example 16 Examples of appositions detected by the patterns

(1) Probenecid is known to interact with the metabolism or renal tubular excretion of *[many drugs]$_{NP}$* (e.g., *[acetaminophen]$_{NP}$*, *[acyclovir]$_{NP}$*, *[theophylline]$_{NP}$*, and *[zidovudine]$_{NP}$*)

(2) Mineral oil interferes with the absorption of *[fat-soluble vitamins]$_{NP}$*, including *[vitamin D preparations]$_{NP}$*.

(3) Epinephrine should not be administered concomitantly with other *[sympathomimetic drugs]$_{NP}$* (**such as** *[isoproterenol]$_{NP}$*) because of possible additive effects and increased toxicity.

(4) It is not known whether other *[progestational contraceptives]$_{NP}$*, **such as** *[implants]$_{NP}$* and *[injectables]$_{NP}$*, are adequate methods of contraception during acitretin therapy.

On the other hand, appositions are usually encoded as hypernymic propositions. In the previous example, the apposition involves a taxonomix relationship between *Catecho-lamine-depleting drugs* (hyperonim) and *reserpine* (hyponym). Therefore, the detection of appositions can also provide very useful information to classify drugs into drug families. For example, Kolarik et al. [2007] defined patterns to capture appositions that represent relations between a hypernym and one or more hyponyms, achieving to detect new drug categories not contained in the DrugBank database.

7.4. Clause splitting

A clause is a group of grammatically-related words that contains a subject and a predicate, though sometimes the subject can be implied. Every sentence consists of one or more clauses [Trask, 1999]. Sentences can be classified by their number and type of clauses. Table 7.6 shows this classification and also some sentences taken from the DrugBank database.

Biomedical texts usually consist of extremely long sentences. Long sentences are usually complex or compound-complex sentences, that is, contain two or more clauses. For example, the last sentence in table 7.6 contains two independent clauses (marked with *indep.clause$_1$* and *indep.clause$_2$*). Both clauses have the same subject: *Coadministration of CRIXIVAN and other drugs that inhibit CYP3A4.* This subject also includes a relative clause *that inhibit CYP3A4* whose subject is *other drugs.* Parsing-based and pattern-based approaches are inefficient to deal with complex and compound sentences. Parsers are usually trained in common English text corpora and are difficult to extend to new domains. For this reason, they usually fail particularly in biomedical complex sentences. Regarding the pattern-based methods, relations are possibly extracted incorrectly when patterns are matched beyond the scope of one clause or other kinds of grammatical units [Huang et al., 2006a]. For example, the previous example contains a relative clause *that inhibit CYP3A4*, which hinders the matching between the sentence and the

Type	Example
Simple sentences, also called **independent clauses**, contain a subject and a verb, and express a complete thought.	Aspirin may decrease the effects of probenecid, sulfinpyrazone, and phenylbutazone.
Compound sentences contain two independent clauses joined by a coordinator.	[Concomitant administration of corticosteroids with aspirin]$_{Subject_{1,2}}$ [may increase the risk of gastrointestinal ulceration]$_{Indep.Clause_1}$ and [may reduce serum salicylate levels]$_{Indep.Clause_2}$.
Complex sentences contain an independent clause joined by one or more subordinate clauses.	[Aspirin is contraindicated in [patients]$_{Subject_{relative}}$]$_{Indep.Clause}$ [who are hypersensitive to nonsteroidal anti-inflammatory agents]$_{RelativeClause}$.
Complex-compound sentences contain at least two independent clauses and one or more subordinate clauses.	[Coadministration of CRIXIVAN and [other drugs]$_{Subject_{Relative}}$]$_{Subject_{1,2}}$ [that inhibit CYP3A4]$_{RelativeClause}$ [may decrease the clearance of indinavir]$_{Indep.Clause_1}$ and [may result in increased plasma concentrations of indinavir]$_{Indep.Clause_2}$.

Table 7.6: Classification of sentences.

pattern *COADMINISTRATION OF <DRUG> AND <DRUG> (MAY)? (DECREASE|INCREASE|...) <EFFECT>*. That is, patterns usually fail in the extraction of relationships from complex and compound sentences.

Clause splitting is the task of dividing a complex or compound sentence into several clauses. This section proposes an algorithm for clause splitting that aims to reduce the complexity of sentences in biomedical texts, in order to improve the performance of our pattern-based method for DDI extraction. The algorithm exploits syntactic and lexical information provided by MMTx. Once sentences have been split into clauses, a set of simplification rules is used in order to generate new independent sentences from the clauses. Finally, the lexical patterns defined by the pharmacist can be applied to the generated sentences in order to extract DDIs.

Figure 7.5 shows an example of a compound sentence containing information on the possible effects of the co-administration of *corticosteroids* and *aspirin*. The sentence consists of two independent clauses, which must be detected by the algorithm. Taking into account that both clauses share the same subject *Concomitant administration of corticosteroids with Aspirin*, two new sentences are generated. Both generated sentences contain the same interaction between *corticosteroids* and *aspirin*, but describe different effects of the interaction. Then, the pharmacological patterns are applied to the generated sentences, in order to extract the interactions and their related information. In this way, the approach does not only extract the interaction between *corticosteroids* and *aspirin*, but also its different effects. If the original sentence is not split into its clauses, the pattern is not able to extract the second effect of the interaction between *corticosteroids* and *aspirin* shown in the original sentence, that is, the reduction of the serum salicylate levels.

We now explain how the sentences are broken into clauses. First of all, it is necessary to ensure that the sentence is actually a compound or a complex sentence, because sometimes the coordinators and subordinators do not function like connectors between clauses, but as prepositions, adverbs, etc. A possible heuristic is to count the number of verb phrases included in the sentence. To give a definition of verb phrase is not an easy task. In fact, linguists have not even reached an agreement on what the verb phrase should include: only the words that are verbs, or also the complements of the verb. While the generative grammarians propose that a verb phrase consists of various combinations of the main verb and any auxiliary verbs, plus optional specifiers, complements, and adjuncts, for functionalist linguists the verb phrases consist only of main verbs, auxiliary verbs, and other infinitive or participle constructions [Calzolari et al., 2001]. Example 17 shows a sentence in which the verb phrase is interpreted in two different ways according to the above definitions.

We have decided to adopt the last definition, that is, we define a verb phrase as a syntactic structure that is composed of a main verb and, optionally, of auxiliary and modal verbs, but the complements are excluded of this structure. Unfortunately, MMTx offers an even simpler definition of verb phrase,
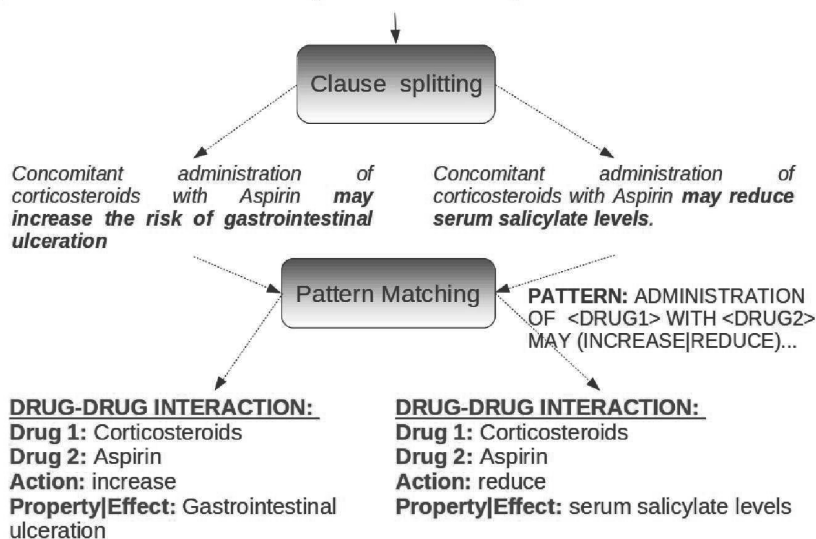
Figure 7.5: Example of clause splitting.

Example 17 What should a verb phrase include?.

Including complements: Anagrelide [may interacts with any of these compounds]$_{VP}$.

Only verbs: Anagrelide [may interacts]$_{VP}$ [with any of these compounds]$_{PP}$.

---

because MMTx labels each verb as a *VP*. Forms of *to be* are labeled as *V/be*. In order to group the main verb, its auxiliary or modal verbs, as well as its adverbial complements in the same verb phrase, we define the *VP-pattern* shown in table 7.7. The *VP-pattern* is applied to sentences in order to merge their adjacent verb phrases into an extended verb phrase. If a sentence contains two or more extended VPs, then we can conclude that it is a complex or compound sentence. However, if a sentence only contains an extended *VP*, it is a simple sentence despite containing any conjunction. First column in table 7.8 shows some sentences parsed by MMTx, while the second column shows the result of applying our *Vp-pattern* to them.

VP-pattern	[VP|V/be|VPG] (V/be)? (NOT)? (ADV)? (VP|V/be|VPG)? (TO VP)?

Table 7.7: VP-pattern: a pattern to attach main verb and its complements.

Once it has been determined that the sentence contains two or more clauses, the following step is to determine the type of sentence. Such information will be very useful in detecting the clause boundaries. In the English language, a *compound sentence* is composed of two or more independent clauses joined by a conjunction that can be a coordinator (coordinating conjunction), a correlative conjunction or an independent marker word (see table 7.9). An independent marker word is a connecting word used at the beginning of an independent clause. Semicolons and commas can also function as conjunctions. If an independent marker occurs at the beginning of the sentence, then a semicolon or a comma should separate the clauses. If the second independent clause starts with an independent marker, then a semicolon or a comma is needed before the marker [Wingersky et al., 2008]. The independent markers can also occur in simple sentences, as in the following sentence: *However, initial dose modification is generally not necessary*. Example 18 shows some compound sentences taken from the DrugBank database.

A *complex sentence* has an independent clause joined with one or more subordinate clauses. A

Verb phrases detected by MMTx	Verb phrases joined by the VP-pattern
[Anagrelide]$_{NP}$ [may]$_{VP}$ [interact]$_{VP}$ [with any of these compounds]$_{PP}$.	[Anagrelide]$_{NP}$ [**may interact**]$_{VP}$ [with any of these compounds]$_{PP}$.
[Its toxicity]$_{NP}$ [may]$_{VP}$ [be]$_{V/be}$ [enhanced]$_{VP}$ [by leucovorin]$_{PP}$	[Its toxicity]$_{NP}$ [**may be enhanced**]$_{VP}$ [by leucovorin]$_{PP}$
[Formal drug interaction studies]$_{NP}$ [have]$_{VP}$ [not]$_{ADV}$ [been]$_{V/be}$ [conducted]$_{VP}$ [with ORENCIA.]$_{PP}$	[Formal drug interaction studies]$_{NP}$ [**have not been conducted**]$_{VP}$ [with ORENCIA.]$_{PP}$
[The combination]$_{NP}$ [of methotrexate]$_{PP}$ [with acitretin]$_{PP}$ [is]$_{V/be}$ [also]$_{ADV}$ [contraindicated]$_{VP}$	[The combination]$_{NP}$ [of methotrexate]$_{PP}$ [with acitretin]$_{PP}$ [**is also contraindicated**]$_{VP}$
[Glucocorticoids]$_{NP}$ [have]$_{VP}$ [been]$_{V/be}$ [shown]$_{VP}$ [to]$_{ADV}$ [reduce]$_{VP}$ [PROLEUKIN-induced side effects]$_{NP}$	[Glucocorticoids]$_{NP}$ [**have been shown to reduce**]$_{VP}$ [PROLEUKIN-induced side effects]$_{NP}$

Table 7.8: How does MMTx label the verb phrases?.

Conjunctions	
Coordinators (*coord*)	for, and, nor, but, or, yet, so.
Correlative conjunctions	[both|either|whether]...or not only...but also
Independent markers (*indMarker*)	However, Moreover, Furthermore, Consequently, Nevertheless, Therefore.
Subordinate conjunctions (*depMarker*).	after, although, as, as if, because, before, even if, even though, if, in order to, since, though, unless, until, whatever, when, whenever, whether, while.

Table 7.9: Conjunctions.

subordinate clause, or dependent clause, is a clause that is embedded as a constituent of a main sentence, acting like a noun, adjective, or adverb in the resulting complex sentence. Subordinate clauses contain both a subject and a verb, but do not express a complete thought. There are two kinds of subordinate clauses: *adverbial clauses* and *relative clauses*. While an adverbial clause acts as an adverb modifying another clause, a relative clause describes the referent of a head noun or pronoun. Relative clauses are embedded in the main clauses by relative pronouns. These pronouns can refer to persons, animals, places, ideas or things (see table 7.11). As a relative pronoun relates to another noun preceding it in the sentence, it connects a dependent clause to an antecedent, that is, a noun that precedes the pronoun. Within the relative clause, the relative pronoun stands for the noun phrase that it references in the main clause (its antecedent), which is one of the arguments of the verb in the relative clause. Therefore, relative pronouns act as the subject or object of the dependent clause. Relative pronouns can be considered as anaphoric expressions, and their resolution was described in detail in chapter 5. Relative clauses are extremely common in biomedical texts.

A complex sentence always has a relative pronoun or a subordinator that links the clauses (see tables 7.9 and 7.11). If the complex sentence begins with a subordinator, that is, the subordinate clause is at the beginning of the sentence, then the subordinate clause should end with a comma. On the other hand, if the independent clause is attached at the beginning of the main sentence and the subordinator is in the middle, then no comma is required [Wingersky et al., 2008]. Example 19 shows some complex sentences taken from the DrugBank database.

Compound sentences	CLAUSE$_1$(,|;)? [**indMarker**|**coord**|;|,] CLAUSE$_2$
	indMarker(,)? CLAUSE$_1$[,|;] CLAUSE$_2$
Complex sentences	**depMarker**(,)? CLAUSE$_{subordinate}$, CLAUSE$_{main}$
	CLAUSE$_{main}$ [**depMarker**|;|,] CLAUSE$_{subordinate}$
Relative Clauses	**relPron** (NP|PP|UNK|ADJ|APOS|COORD)? VP [NP|PP|UNK|ADJ|APOS|COORD]

Table 7.10: Initial patterns for clause splitting

Taking into account the above clues, we initially defined a set of lexical patterns for detecting clauses boundaries in compound and complex sentences. These patterns are shown in table 7.10. If a sentence

Example 18 Compound sentences.

(1) [*Erythromycin and clarithromycin (and possibly other macrolide antibiotics) and tetracycline may increase digoxin absorption in patients who inactivate digoxin by bacterial metabolism in the lower intestine*]$_{clause_1}$, **so** [*that digitalis intoxication may result*]$_{clause_2}$.

(2) *Potential for drug interactions exists* **not only** *with concomitant medication* **but also** *with drugs administered after discontinuation of amiodarone.*

(3) [*No drug interaction studies have been conducted for COLAZAL*]$_{clause_1}$, **however** [*the use of orally administered antibiotics could, theoretically, interfere with the release of mesalamine in the colon*$_{clause_2}$.

(4) [*Concomitant use of prophylactic low dose heparin did not appear to affect safety*]$_{clause_1}$, **however**, [*its effects on the efficacy of Xigris have not been evaluated in an adequate and well-controlled clinical trial*]$_{clause_2}$.

Example 19 Complex sentences.

(1) [**When** *amiloride HCl is administered concomitantly with an angiotensin-converting enzyme inhibitor,*]$_{SubordinateClause}$ [*the risk of hyperkalemia may be increased.*]$_{MainClause}$.

(2) [*Lithium generally should not be given with diuretics*]$_{MainClause}$ **because** [*they reduce its renal clearance and add a high risk of lithium toxicity.*]$_{SubordinateClause}$.

(3) *The extent [to* **which** *SSRI-TCA interactions may pose clinical problems*]$_{RelativeClause}$ [*will depend on the degree of inhibition and the pharmacokinetics of the SSRI involved.*]$_{MainClause}$.

(4) [*Population pharmacokinetic analyses revealed*]$_{MainClause}$ [**that** *MTX, NSAIDs, corticosteroids, and TNF blocking agents did not influence abatacept clearance.*]$_{RelativeClause}$.

(5) [*Since the excretion of oxipurinol is similar to that of urate*]$_{MainClause}$, [*uricosuric agents,* [**which** *increase the excretion of urate*]$_{RelativeClause}$, *are also likely to increase the excretion of oxipurinol and thus lower the degree of inhibition of xanthine oxidase*].

Relative Pronouns	Grammatical Relation	Referring to	Num
who	subject or object	people	101
whoever	subject	people	0
whom	object, especially in non-defining relative clauses	people	12
whomever	object	people	0
whose	possession	people, animals and things	9
which	subject or object	animals and things. Also, it can refer to a whole sentence	142
that	subject or object	people, animals and things in defining relative clauses; who or which are also possible	569
whichever,whatever		more than one place, thing or idea	0

Table 7.11: Distribution of relative pronouns in the DrugDDI corpus.

matches some of these patterns, then its clauses can be easily extracted from the matching. Relative clauses are a especial case, since, they often appear in the middle of a main clause, splitting it into two parts. For example, the last sentence in Example 19 contains a relative clause *which increase the excretion of urate*, which separates the main sentence in two different parts. To separate such clause from the original sentence, its boundaries must be determined. A special pattern for identifying relative clauses has been defined. Relative clauses begin with a relative pronoun and contain a subject (though sometimes the pronoun itself can be the subject) and a verb with its complements. However, these patterns are not always enough. Determining where a clause ends is not always a trivial task, since there might be commas or conjunctions internal to the clause. Moreover, some conjunctions can also function as prepositions (for example *for*) or as adverbs (for example *yet*, *so*). The problem regarding adverbs is easily resolved (at least in most of cases) because MMTx labels them as *CONJ* phrases when they function as coordinators (though sometimes MMTx mistakes the phrases or is not able to determine the types). The previous identification of appositions and coordinate structures (see sections 7.3 and 7.2) allows to reduce the number of commas and conjunctions internal to a clause. However, for each comma or coordinator not included in any apposition or coordinate structure, it is required to know whether the

Example 20 Where does the clause end?.

(1) *[These reactions consisted of [erythema, pruritus, and hypotension]$_{COORD_1}$]$_{clause_1}$* **and** *[occurred within hours of administration of chemotherapy]$_{clause2}$.*
In this example, the previous detection of the coordinate structure 'erythema, pruritus, and hypotension' allows to correctly identify the second 'and' as the coordinator that joins the two independent clauses. The subject (*These reactions*) is ommited in the second clause.

(2) *[[[Intestinal adsorbents (charcoal)]$_{APOS_1}$ and [carbohydrate-splitting enzymes (e. g., amylase, pancreatin)]$_{APOS_2}$]$_{COORD_1}$] may reduce the effect of Acarbose]$_{clause_1}$* **and** *[should not be taken concomitantly]$_{clause_2}$.*
In this sentence, the previous identification of the two appositive structures and the coordinate structure between them allows to identify the second 'and' as a coordinator between the two clauses: $clause_1$ and $clause_2$. In this sentence, the subject of the second clause is also omitted ($COORD_1$).

(3) *[An increase in serum lithium concentration has been reported during concomitant administration of lithium with ATACAND]$_{clause_1}$, **so** [careful monitoring of serum lithium levels is recommended during concomitant use]$_{clause_2}$.*
In this sentence, the two conjoined clauses are separated by the conjunction 'so'. MMTx labels this words with the *CONJ* type, and this label easily allows to identify the boundaries between the two clauses.

(5) *[The biochemical activity of debrisoquin hydroxylase is reduced in a subset of the caucasian population]$_{clause_1}$ [(about 7-10% of caucasians are **so** called poor metabolizers)]$_{clause_2}$*
However, although the sentence contains two clauses (the second clause is marked by commas), the word 'so' in this sentence functions as an adverb (classified as *ADV* phrase by MMTx), and not as conjunction.

---

clause ends or not in it. Therefore, the above patterns have been replaced with a set of heuristics based on the observation of fifty compound and complex sentences from the training set. These heuristics are encoded in algorithm 1.

Algorithm 1 Clause splitting in a compound or complex sentence S

Require: S!=NULL and its verbs have been joined into VPs by the VP-pattern.
    {S is a sentence.}
 1: Define $NUMVP$ as the number of verb phrases in S.
 2: **if** $NUMVP==1$ **then**
 3:     S is a simple sentence {S only contains a indepedent clause.}
 4:     **return**
 5: **end if**
 6: INI:=0. {This is the position where S begins}
 7: Look for a separator marker from INI in S, that is, a coordinator, a subordinator, an independent marker, a semicolon or a comma
    {The coordinator or independent marker must be classified as a *CONJ* phrase by MMTx.}
 8: Save the found marker into the variable $MARKER$.
 9: Define FIN as the position where $MARKER$ begins.
10: **while** $MARKER$!=NULL **do**
11:     Define $CLAUSE$ as the substring between INI and FIN.
12:     **if** $CLAUSE$ has any VP **then**
13:         Mark $CLAUSE$ as a clause in S. {The algorithm has found a clause. It must continue with the search of the rest of clauses}.
14:         Initialize $CLAUSE$ to NULL.
15:         To re-define INI as the position where $MARKER$ ends.
16:     **else**
17:         Look for a separator marker from FIN in S.
18:     **end if**
19:     Save the found marker into the variable $MARKER$.
20:     Define FIN as the position where $MARKER$ begins.
21: **end while**
22: **if** CLAUSE!=NULL **then**
23:     Mark $CLAUSE$ as a clause.
24: **end if**

The input of the algorithm is the sentence in which its verb phrases have been joined by the VP-pattern. First of all, the algorithm must check that the sentence contains two or more clauses. Then, the sentence is reviewed while it contains any separator marker. A separator marker can be a coordinator, a independent marker, a dependent marker, a semicolon or a comma. The coordinators and subordinators must be labeled by MMTx as *CONJ* phrases, otherwise, they are not considered as conjunctions. Then, the algorithm iteratively finds candidate clauses, that is, a substring of the sentence between markers. If the candidate clause contains a verb phrase, then it is considered as clause. The algorithm is able to decide the kind of clause, that is, independent or subordinate. A clause is subordinate, if it begins with a subordinate conjunction and ends with a comma, or, if the clause is attached at the end of a main clause.

7.4.1. Rules for Sentence Simplification

Once appositions and coordinate propositions have been recognized, and compound and complex sentences have been split into clauses, it is possible to apply a set of rules for sentence simplification. These rules allow to simplify the complex and compound sentences in simple sentences. Then, the pattern-based approach for DDI extraction will be applied to these simpler sentences. We have adapted some of the simplification rules presented in [Siddharthan, 2006]. This approach also recognizes relative clauses, apposition, coordination and subordination, but its goal is not the relation extraction, but to provide syntactic simplification of sentences. "Syntactic simplification is the process of reducing the grammatical complexity of a text, while retaining its information content and meaning" [Siddharthan, 2006]. This process can improve the performance of several NLP applications such as text summarization or machine translation. Siddharthan [2006] proposes seven simplification rules to generate new simplified sentences from the clauses of the complex and compound sentences. We have only adapted three of these rules. Table 7.12 presents the rules adapted in our approach, and examples 21 and 22 contain some sentences broken up into simpler sentences by these rules. The clause $CLAUSE_{REL(NP)}$ means that it is attached to the noun phrase NP.

Simplification Rules	Generated sentences
MARKER(,)? $CLAUSE_1$, $CLAUSE_2$	(1) $CLAUSE_1$ (2) $CLAUSE_2$
$CLAUSE_1(,)?$ **MARKER** $CLAUSE_2$	(1) $CLAUSE_1$ (2) $CLAUSE_2$
$CLAUSE_1$ NP $CLAUSE_{REL(NP)}$ $CLAUSE_2$	(1) $CLAUSE_1$ NP $CLAUSE_2$ (2) NP $CLAUSE_{REL(NP)}$

Table 7.12: Rules to generate new simplified sentences from the clauses.

Example 21 Simplification of complex and compound sentences

(1) [Because]$_{MARKER}$ [busulfan is eliminated from the body via conjugation with glutathione]$_{CLAUSE_1}$ [use of acetaminophen prior to (72 hours) or concurrent with BUSULFEX may result in reduced busulfan clearance based upon the known property of acetaminophen to decrease glutathione levels in the blood and tissues]$_{CLAUSE_2}$.
(2) [Although]$_{MARKER}$ [the interactions observed in these studies do not appear to be of major clinical importance$_{CLAUSE_1}$, [BREVIBLOC should be titrated with caution in patients being treated concurrently with digoxin, morphine, succinylcholine or warfarin.]$_{CLAUSE_2}$
(3) [Trimeprazine also decreases the effect of heparin and oral anticoagulants,]$_{CLAUSE_1}$ [while]$_{MARKER}$ [MAOIs can increase the effect of trimeprazine.]$_{CLAUSE_2}$

Example 22 The following sentence (containing a relative clause) is transformed into the two simpler sentences (1) and (2)

Since the excretion of oxipurinol is similar to that of urate, uricosuric agents, *which increase the excretion of urate*, are also likely to increase the excretion of oxipurinol and thus lower the degree of inhibition of xanthine oxidase.
(1) Since the excretion of oxipurinol is similar to that of urate, uricosuric agents are also likely to increase the excretion of oxipurinol and thus lower the degree of inhibition of xanthine oxidase.
(2) Uricosuric agents (which) increase the excretion of urate.

7.5. Evaluating syntactic structures resolution

The evaluation of appositions and coordinate structures resolution was performed on a set of fifty sentences from the training corpus. The sentences were randomly selected and manually tested with the assistance of a linguist. Results are shown in tables 7.13 and 7.14.

Structure	TP	FN	FP	P	R	F
Coordinate	24	14	0	1	0,63	0,77
Appositions	11	2	0	1	0,85	0,92

Table 7.13: Evaluation of appositions and coordinate structures resolution.

We observed that most of the errors were due to tagging and parsing mistakes made by MMTx. Several studies have dealt with the error analysis of MMTx. Both the error analysis and the improvement of MMTx are two issues that are out of scope of this thesis. On the other hand, we are aware that this evaluation is quite shallow to reach definite conclusions about performance. Future directions include increasing the size of the corpus, trying to identify and resolve the errors of MMTx in order to improve the appositive and coordinate structures resolution, and resolving new kinds of constructions not addressed in the current approach like the kind of apposition shown in example 23. In addition, we are planning to study the utility of the Genia-GR [Tateisi et al., 2008] corpus in the evaluation, which consists of 50 abstracts and is annotated with grammatical relations.

Example 23 Appositions not linked by any marker.

Concomitant use of *argatroban, an anticoagulant* with antiplatelet agents may increase the risk of bleeding.
A possible interaction between glyburide and *ciprofloxacin, a fluoroquinolone antibiotic* has been reported, resulting in a potentiation of the hypoglycemic action of glyburide.

Regarding the evaluation of the clause boundaries detection, it was also performed on the aforesaid set of sentences. The results are shown in table 7.14.

Structure	TP	FN	FP	P	R	F
Relatives	6	3	0	1	0,67	0,8
Rest of Clauses	16	7	0	1	0,7	0,82

Table 7.14: Results of clause splitting.

Clause splitting is a very complex task, which consists of three tasks: identifying clause starts, identifying clause ends, and finding complete clauses (many of them may be nested clauses). The nesting of clauses is very common in biomedical texts. Our method mainly fails to deal with the resolution of nested clauses (see example 24). However, though it obtains lower results, we believe that it is a good initial approximation for clause splitting in the biomedical domain. As it is obvious, the simplification rules also fail when the clause splitting is wrong. Future work should include providing a more detailed error analysis of our method for clauses splitting and simplification rules in order to improve the results, evaluating it on the Penn Treebank used in CoNLL 2001 shared task [Tjong et al., 2001], and annotating a small set of our corpus with clause boundaries to apply machine learning methods. These methods have been widely applied to the clauses splitting [Carreras and Marquez, 2005, Nguyen et al., 2009] achieving good results, because they are able to learn from a relatively small annotated corpus.

7.6. The set of lexical patterns to extract DDIs

Despite the richness of natural language expressions, in practice, DDIs are often expressed by a limited number of constructions. This fact favors the use of patterns as an excellent method for their extraction. Based on the observation of the training set, our pharmacist defined a set of lexical patterns (see table 7.15) to capture the various language constructions used to express DDIs in pharmacological texts. Moreover, the pharmacist provided synonyms for the verbs shown in table 7.16 that can indicate a possible interaction between drugs.

Example 24 Nested clauses

[Coadministration of CRIXIVAN and [other drugs that inhibit CYP3A4]] [may decrease the clearance of indinavir] and [may result in increased plasma concentrations of indinavir].

Id	Pattern
P1	DRUG *MODAL*? *ADV*? (INTERACT|INTERFERE) WITH WORD$_{0..5}$ (OF)? DRUG
P2	DRUG *MODAL*? *ADV*? INCREASE$_{syn}$ WORD$_{0..5}$ (OF)? DRUG
P3	DRUG *MODAL*? *ADV*? DECREASE$_{syn}$ WORD$_{0..5}$ (OF)? DRUG
P4	DRUG *MODAL*? *ADV*? ALTER$_{syn}$ WORD$_{0..5}$ (OF)? DRUG
P5	DRUG *MODAL*? BE *ADV*? INCREASE$_{syn}$ WORD$_{0..5}$ (BY)? DRUG
P6	DRUG *MODAL*? BE *ADV*? DECREASE$_{syn}$ WORD$_{0..5}$ (BY)? DRUG
P7	DRUG *MODAL*? BE *ADV*? ALTER$_{syn}$ WORD$_{0..5}$ (BY)? DRUG
P8	COADMINISTRATION OF DRUG (WITH|AND|PLUS) DRUG *MODAL*? *ADV*? [INCREASE$_{syn}$|DECREASE$_{syn}$INTERACT$_{syn}$||ALTER$_{syn}$]
P9	COADMINISTRATION OF DRUG (WITH|AND|PLUS) DRUG *MODAL*? *BE*? *ADV*? RESULT$_{syn}$ (TO|WITH|IN) [INCREASE$_{syn}$|DECREASE$_{syn}$INTERACT$_{syn}$||ALTER$_{syn}$]
P10	CAUTION *MODAL*? *ADV*? *BE*? USED WHEN DRUG *WORD*? (WITH|AND|PLUS) DRUG *BE*? ADMINISTERED *CONCURRENTLY*?
P11	PATIENTS TREATED (WITH)? DRUG (WITH|AND|PLUS) DRUG (CONCURRENTLY)? MODAL BE OBSERVED
P12	INTERACTION (OF|BETWEEN) DRUG (AND|WITH|PLUS) DRUG MODAL? (BE)? WORD$_{0..3}$ (OBSERVED|INCREASE|DECREASE|ALTER)

Table 7.15: Lexical patterns to extract DDIs.

MODAL=[CAN|COULD|MAY|MIGHT|SHOULD|MUST|HAVE|HAS|HAD]
BE=[IS|ARE|WAS|WERE|BE|BEEN]
ADV is any adverbial except 'NOT'. For example, also,potentially, etc.
INCREASE$_{syn}$=[ELEVATE|ENHANCE|EXACERBATE|EXTEND|INCREASE|INTENSIFY|POTENTIATE|PROMOTE|PROLONG| RAISE|RISE|STIMULATE]
DECREASE$_{syn}$=[AUGMENT|ELEVATE|ENHANCE|EXACERBATE|EXTEND|GO_UP|INCREASE|INTENSIFY|POTENTIATE| PROMOTE|PROLONG|RAISE|RISE|STIMULATE]
ALTER$_{syn}$=[ACCELERATE|ANTAGONIZE|ALTER|CHANGE|INDUCE|INFLUENCE|INHIBIT|INTERFERE]
RESULT$_{syn}$=[RESULTS|ASSOCIATED|SHOWN|RESULTED|OBSERVED|DETERMINED]
< **DRUG PROPERTIES** >=[PROPERTY|EFFECT|. . .]
WHEN=[WHEN|IF|WHETHER]
ADMINISTERED=[CO-ADMINISTERED|COADMINISTERED|ADMINISTERED|TAKEN|GIVEN|USED|EMPLOYED]
PATIENTS=[PATIENTS|SUBJECTS]
TREATED=[TAKEN|TREATED|RECEIVING|TAKING]

Table 7.16: Auxiliary patterns

7.7. Evaluation

This section explains in detail the experiments that we have carried out. First of all, we describe our baseline experiment, the most basic experiment in which neither coordinations, appositions nor clauses are tackled, that is, the pharmacological patterns are directly applied to the text of sentences. The following steps summarize this process:

Algorithm 2 Baseline procedure: patterns are directly matched against sentences.

1: The text is split into sentences to separately treat each of them.
2: Each sentence is parsed by MMTx providing lexical information, POS tags, syntactic types, and semantic information on its words, tokens and phrases.
3: The DrugNer (chapter 4) identifies the drug names in sentence.
4: Select those sentences that contain two or more drug names.
5: Replace the drug names by the label $DRUG._{index}$, where $index$ shows the order of each drug in the list of drugs which occur in the sentence. For example, if the sentence contains three drugs, their names will replaced with $DRUG._1$, $DRUG._1$ and $DRUG._3$ tags.
6: The set of pharmacological patterns is applied to the text of the sentence.
7: When a sentence has been correctly matched with a pattern, it must be checked if the matching string includes the negative adverb (NOT). If it is not included, then a possible interaction has been found.
8: Drug names that occur in the matching are retrieved, and the pair of drug names is proposed as a drug-drug interaction.
9: Check that the pair of corresponding phrases is annotated as a DDI in the DrugDDI corpus in order to evaluate it.

Id	TP	FP	FN	P(%)	R(%)	$F_{\beta\,=\,1(\%)}$
P1	17	11	3010	60.71	0.56	1.11
P2	114	50	2913	69.51	3.77	7.15
P3	65	57	2962	53.28	2.15	4.13
P4	81	37	2946	68.64	2.68	5.15
P5	19	5	3008	79.17	0.63	1.25
P6	9	6	3018	60.00	0.30	0.59
P7	24	7	3003	77.42	0.79	1.57
P8	15	0	3012	100.00	0.50	0.99
P9	31	11	2996	73.81	1.02	2.02
P10	6	1	3021	85.71	0.20	0.40
P11	29	4	2998	87.88	0.96	1.90
P12	16	18	3011	47.06	0.53	1.05
$GLOBAL$	426	207	2601	67.30	14.07	23.28

Table 7.17: Basic Experiment Results. TP=True Positive, FP=False Positive, P=Precision, R=Recall.

Table 7.17 shows the global and individual pattern performance of this basic experiment. The baseline experiment achieves a reasonable precision (67.30%), but very low recall (14.07%). The average number of DDIs detected by each pattern is 35.5 (the number total of DDIs in the DrugDDI corpus is 3,160). Regarding the individual pattern performance, the highest recall is achieved by the pattern *P2* and the highest precision by the pattern *P8*.

In the second experiment, appositions and coordinate structures are identified in text by the set of syntactic patterns described in sections 7.3 and 7.2. The pharmacological patterns have been modified to consider these structures, that is, they are extended for including the labels *APPOSITION* and *COORD* as possible elements participating in the interactions. Thus, for this experiment, $DRUG :=$ [*DRUG*|*APPOSITION*|*COORD*]. The procedure of matching pattern for this experiment is explained in algorithm 3.

Figures 7.6 and 7.7 show two sentences which are interpreted by the syntactic patterns and matched against the lexical patterns in order to extract DDIs.

Example 25 Examples of extended patterns to include appositions and coordinate structures as possible interacting elements.

P1 (DRUG|APPOSITION|COORD) *MODAL*? *ADV*? (INTERACT|INTERFERE) WITH WORD$_{0..5}$ (OF)? (DRUG|APPOSITION|COORD)
P2 (DRUG|APPOSITION|COORD) *MODAL*? BE *ADV*? INCREASE$_{syn}$ WORD$_{0..5}$ (BY)? (DRUG|APPOSITION|COORD)

Algorithm 3 Pattern Matching including the detection of appositions and coordinate structures

1: The text is split into sentences. Each sentence is treated separately.
2: Each sentence is parsed by MMTx providing lexical information, POS tags, syntactic types, and semantic information on its words, tokens and phrases.
3: The DrugNer (chapter 4) identifies the drug names in the sentence.
4: Select those sentences that contains two or more drug names.
5: The shallow syntactic information provided by MMTx is used to generate a sequence of the syntactic types of the phrases in the sentence.
6: The syntactic patterns are applied to the sequence in order to detect its appositions and coordinate structures. If some structure is detected, this will be replaced with the label *APPOSITION.$_{index}$* or *COORD.$_{index}$*, depending on the case.
7: The drug names are replaced by the label *DRUG.index*
8: The text of the sentence is generated by concatenating their text phrases (except the text of the appositions and coordinate structures).
9: If generated text is matched by some pattern and the matching string does not contain the negative adverb, a candidate interaction has been found.
10: If the matching string contains appositions or coordinate structures, these must be unfolded in order to obtain the individual interacting elements (as many as the number of elements which make up each structure) and build the list of interactions.
11: The list of interactions is evaluated on the DrugDDI corpus.

Table 7.18 shows the global and individual pattern performance of the second experiment. Recall is improved by the inclusion of the appositions and coordinate structures, however, precision is lower. The average number of DDIs detected by each pattern is 64.83. The pattern *P2* still achieves the highest recall, and the highest precision is obtained by the pattern *P10*. Therefore, the detection of these structures achieves to improve the recall (almost 12%) with a significant decrease in precision of almost 19%. This decrease can be attributed to the errors introduced during the syntactic processing.

Id	TP	FP	FN	P(%)	R(%)	$F_{\beta = 1(\%)}$
P1	71	49	2956	59.17	2.35	4.51
P2	212	175	2815	54.78	7.00	12.42
P3	119	147	2908	44.74	3.93	7.23
P4	138	124	2889	52.67	4.56	8.39
P5	40	43	2987	48.19	1.32	2.57
P6	13	20	3014	39.39	0.43	0.85
P7	30	20	2997	60.00	0.99	1.95
P8	27	20	3000	57.45	0.89	1.76
P9	60	28	2967	68.18	1.98	3.85
P10	11	4	3016	73.33	0.36	0.72
P11	38	155	2989	19.69	1.26	2.36
P12	19	35	3008	35.19	0.63	1.23
GLOBALS	778	820	2249	48.69	25.70	33.64

Table 7.18: Results for extended patterns with appositions and coordinate structures.

We now explain the last experiment that combines the detection of appositions, coordinate structures, clause splitting and simplification rules. First of all, appositions and coordinate clauses are detected by applying the previous described procedure (3) step by step until the sixth step. Then, the algorithm 1 (described in section 7.4) is applied to sentences in order to split the complex and compound sentences into their clauses. Then, new sentences are generated from these clauses by the simplification rules described

Allopurinol interacts with **anisindione, azathioprine and cyclosporine**
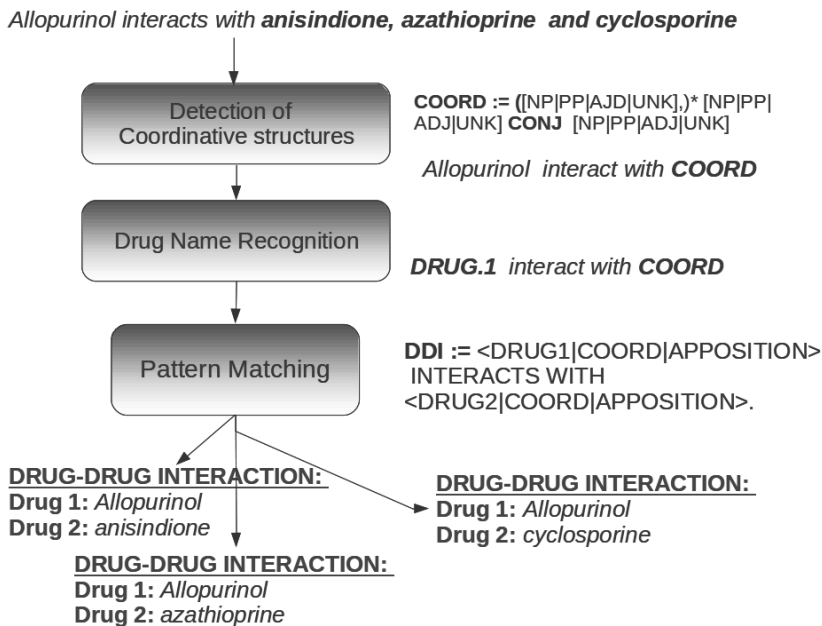
Figure 7.6: Matching procedure

in subsection 7.4.1. Finally, the previous procedure of matching pattern (algorithm 3) is applied to these new sentences from the seventh step. The results are shown in table 7.19. While the inclusion of appositions and coordinate structures achieved to improve the recall, and therefore, the f-measure, the detection of clauses has not improved the performance. This is mainly due to the fact that many interactions occurring in complex sentences often span several clauses. The lexical patterns are not able to capture these interactions. Some examples of this kind of sentences are shown in example 26.

As it has been aforementioned, the errors introduced during the MMTx analysis negatively affect the results obtained with our approach. Also, we are aware that our clause splitting method is too simplistic to deal with the complexity of biomedical sentences. Another shortcoming of our approach is that the negation has been addressed by an heuristic too simplistic. So, the sentence shown in example 28 matches the pattern *P1*, however, it does not represent any interaction. This is due to the previous negation *studies have not shown* has not been detected. This could be avoid by a deeper treatment of the negation. Finally, the richness of natural language causes that our patterns are not enough for identifying many of the interactions (see examples 27).

Example 26 DDIs spanning several clauses

[The Cmax of *norethindrone* was 13% higher] when [it was coadministered with *gabapentin*]
[*Serum theophylline* concentrations increase] when [*grepafloxacin* is initiated in a patient maintained on *theophylline*].
Therefore, [when *MIDAMOR* and *non-steroidal anti-inflammatory agents* are used concomitantly], [the patient should be observed closely to determine if the desired effect of the diuretic is obtained].
[When *such drugs* are withdrawn from a patient receiving *MICRONASE*], [the patient should be observed closely for hypoglycemia.]

Catecholamine-depleting drugs, such as reserpine, *may have an additive effect when given with beta-blocking agents.*

Detection of Appositions

APPOSITION := <APPOSITIVE> MARKER <APOSITIVE>

APPOSITION *may have an additive effect when given with* **DRUG.**

Pattern Matching

DDI := <DRUG1|APPOSITION> (HAVE|INCREASE|...) <EFFECT> WHEN GIVEN WITH <DRUG2|APPOSITION>.

DRUG-DRUG INTERACTION:
Drug 1: *Catecholamine-depleting drugs*
Drug 2: *beta-blocking agents*
Property|Effect: additive

DRUG-DRUG INTERACTION:
Drug 1: Reserpine
Drug 2: beta-blocking agents
Property|Effect: additive
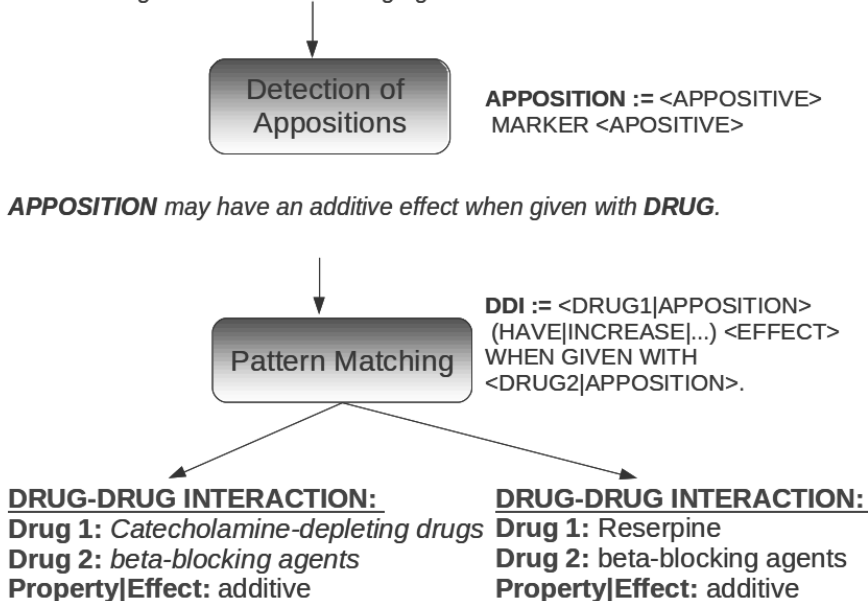
Figure 7.7: Example of sentence with an apposition.

Id	TP	FP	FN	P(%)	R(%)	$F_{\beta = 1(\%)}$
P1	71	49	2956	59.17	2.35	4.51
P2	194	154	2833	55.75	6.41	11.50
P3	121	141	2906	46.18	4.00	7.36
P4	137	123	2890	52.69	4.53	8.34
P5	39	36	2988	52.00	1.29	2.51
P6	9	14	3018	39.13	0.30	0.59
P7	28	20	2999	58.33	0.93	1.82
P8	31	28	2996	52.54	1.02	2.01
P9	60	28	2967	68.18	1.98	3.85
P10	3	3	3024	50.00	0.10	0.20
P11	39	154	2988	20.21	1.29	2.42
P12	19	35	3008	35.19	0.63	1.23
GLOBALS	751	785	2276	48.89	24.81	32.92

Table 7.19: Results of the patterns applied to the clauses

7.8. Conclusions

In this section, we have proposed a hybrid method that combines the resolution of complex linguistic constructions and matching pattern.

Regarding the resolution of the linguistic constructions, as it was pointed out in section 7.5, most of the errors are due to mistakes introduced in the MMTx level and the difficulty of resolving nested clauses, so frequent in biomedical texts. Future directions include trying to identify and resolve the errors of MMTx, improving our clause splitting algorithm, proposing new suitable simplification rules

Example 27 Patterns are not enough to identify some interactions

In a pharmacokinetic substudy in patients with congestive heart failure *receiving furosemide or digoxin in whom therapy with FLOLAN was initiated, apparent oral clearance values for furosemide (n = 23) and digoxin (n = 30) were decreased by 13% and 15%, respectively,* on the second day of therapy and had returned to baseline values by day 87.

In subjects who had received *21 days of 40 mg/day racemic citalopram, combined administration of 400 mg/day cimetidine for 8 days resulted in an increase in citalopram AUC and Cmax of 43% and 39%,* respectively.

Example 28 Negation of interaction.

While studies <u>have not shown</u> *DRUG. interact with DRUG.*, caution should be exercised, nontheles, since interactions have been with drug.

---

to regenerate the simple sentences from clauses, checking what occurs if the resolutions are applied in a different order, studying the utility of other corpora such as Genia-GR or Penn Treebank in the evaluation, and increasing the size of the corpus and annotating it with these linguistic constructions in order to apply machine learning methods.

Concerning the performance in the extraction of DDIs, the variability of natural language expression makes it difficult for our method to accurately detect all semantic relations occurring in text since sentences conveying the same relation may be composed lexically and syntactically differently. Inversely, sentences that are lexically common may not necessarily convey the same relation. Thus, our lexical patterns are not enough to identify many of the interactions. Future work will include application of the SPINDEL system [de Pablo-Sánchez and Martínez, 2009] to semi-automatically learn new patterns from biomedical texts. SPINDEL is a bootstrapping method which has achieved good results for named entity recognition task in general domain. In addition, we will carry out a more exhaustive treatment of negation and modality in sentences.

Chapter 8

Using a kernel-based approach for Drug-Drug Interaction extraction

8.1. Introduction

The diversity of natural language makes relation extraction a difficult task. In the biomedical domain, this task is even more complicated due to the complexity of biomedical sentences. Pattern-based approaches are not able to capture the semantic relations when there are discontinuous word patterns or long-distance dependencies among the entities in a sentence [Kim et al., 2008]. As it was seen in chapter 7, our pattern-based approach obtains relatively low performance. The best results achieved a recall of 25,7% and a precision of 48,7%. Pattern-based approaches achieve low recall rates because lexical-syntactic patterns are not enough to detect all semantic relations occurring in text, since sentences containing the same semantic relation can be lexically and syntactically different. Inversely, sentences that are lexically and syntactically similar, may not necessarily contain the same relation. The objective is to propose an alternative approximation for the extraction of drug-drug interactions that overcomes the shortcomings of our pattern-based approach and achieves to improve its results.

Recent studies on relation extraction have shown the advantages of machine learning approach to this problem [Zhang et al., 2008, Giuliano et al., 2007b, Bunescu and Mooney, 2005, Jiang and Zhai, 2007, Culotta and Sorensen, 2004, Zelenko et al., 2003]. In machine learning, we can distinguish two important paradigms: feature-based and kernel-based systems. In the first paradigm, a set of features is carefully selected from different levels of text analysis such as tokenization, part-of-speech tagging or syntactic analysis. However, the huge amount of words may produce high or even infinite dimensionality of the feature space becoming computationally infeasible. In addition, features are not able to correctly capture the structural information from complex structures such as parse trees or dependency graphs. This type of representation based on features has been widely used [Kambhatla, 2004, GuoDong et al., 2005, Jiang and Zhai, 2007]. The second paradigm proposes the use of the structural representations (such as sequences, parse trees or dependency graphs) as an alternative to the set of features, offering a good solution to the shortcomings of the feature-based paradigm. Kernel functions are designed on some structured representation of the relation instances to capture the similarity between two relation instances, thus preserving important structural information and without need to explicitly define a set of features. Recently, these kernel methods have been successfully applied to the detection of semantic relations in both general and biological domains [Bunescu and Mooney, 2005, Zhang et al., 2008, Kim et al., 2008].

We can formulate the extraction of DDIs as a binary classification problem. Given a sentence,

$$S = w_1 w_2 .. DRUG_1 .. DRUG_2 .. w_n \tag{8.1}$$

the objective is to find a function F that is able to decide if the drugs $DRUG_1$ and $DRUG_2$ contained in the sentence S express a DDI:

$$F(T(S)) = \begin{cases} 1 & \text{if } DRUG_1 \text{ and } DRUG_2 \text{ interact} \\ 0 & \text{otherwise} \end{cases} \tag{8.2}$$

F can be a discriminative classifier such as Support Vector Machine, Voted Perceptron, Log-linear model, among others. $T(S)$ is the representation of the sentence S, that can be a set of features extracted

from S, or a structured representation of the sentence. If S is represented by a set of features extracted from it, then, we can use a similarity function like cosine distance to compare its representation with the set of positive and negative examples, and so determine if its drugs interact. If a structured representation is used, then it is necessary to define a similarity metric, that is a kernel function, in order to estimate the similarity between the structured representation of the sentence and the structured representations of the positive and negative relation instances (examples).
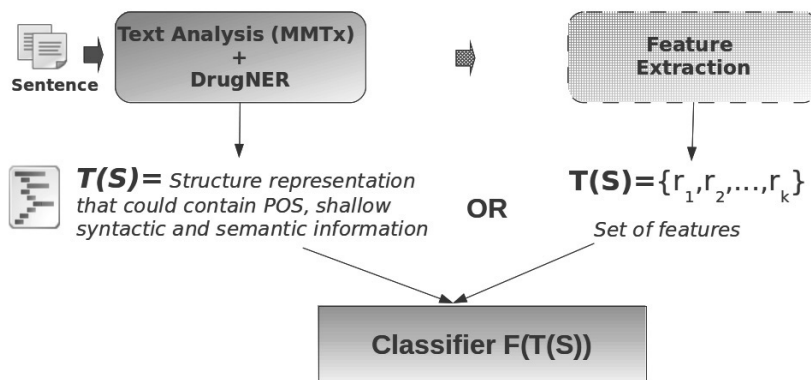
Figure 8.1: Sentence representation: Structure-based representation vs feature-based representation.

Parse trees are a natural representation that allows to directly capture the structural information within sentences. Several approaches [Zelenko et al., 2003, Li et al., 2008] have shown that tree kernels not only outperform feature-based methods, they also achieve better results than sequence kernels. However, tree kernels are relatively slow compared to feature classifiers and sequence kernels [Bunescu and Mooney, 2005, Li et al., 2008]. The complexity of tree kernels and the need to evaluate thousands of them during the process of classification may render them inappropriate for practical purposes. For this reason, we believe that sequence kernels are more appropriate than tree kernels for the extraction of drug-drug interactions since these methods should be integrated into a real application in which the processing time will be a priority.

In this chapter, we describe our second approximation for DDI extraction based on the kernel-based approach presented by Giuliano et al. [2006]. This approach combines two sequence kernel methods to integrate the information of the whole sentence where the relation occurs and the context information about the interacting entities. Each relation instance is represented as a sequence of words taking into account the sequential order of words in the sentence. The approach only needs shallow syntactic information such as sentence splitting, tokenization, part-of-speech (PoS) tagging and lemmatization to build the structured representations of relation instances. We have used the java tool for relation extraction (jSRE)[1], which is based on Giuliano et al. [2006]'s approach. jSRE has been used for relation extraction with good results in both general and biological domains [Giuliano et al., 2006, 2007b]. The fact that this tool only needs shallow syntactic information is crucial for us since we have analyzed texts using the biomedical parser MMTx, which only provides shallow syntactic information. General-purpose syntactic parsers such as [Collins, 1996] or [Charniak, 2000] are not able to deal with the complexity of the biomedical sentences. There are no syntactic parsers trained on biomedical texts, except Genia dependency parser [Sagae, 2007] and the parser presented in [Lease and Charniak, 2005]. We decided to use MMTx instead of the aforesaid parser because MMTx also provides semantic information that can be exploited to identify drug names as well as other biomedical concepts. As it will be explained later, the information provided by MMTx is used to represent each relation instance as a sequence of words, PoS tags and the tag *DRUG* (drug names are labeled with this label).

The chapter is organized as follows: the method proposed by Giuliano et al. [2006] is described in detail in section 8.2. Section 8.3 describes the experiments conducted. Finally, section 8.4 draws some conclusions and unresolved issues, and suggests directions for future work.

---

[1]http://tcc.itc.it/research/textec/tools-resources/jsre.html

8.2. A shallow syntactic kernel for relation extraction

This section describes the kernel-based approach proposed in [Giuliano et al., 2006]. As it was explained in chapter 6, kernels-based approaches [Shawe-Taylor and Cristianini, 2004] provide an effective alternative to feature-based approaches. The main advantages of these methods is that they can exploit the structural descriptions of words, phrases and sentences, and process them efficiently. Kernels do not need to represent each data instance into a flat set of features, but just define a similarity measure that determines the similarity between instances. The intuitive idea behind a kernel method is to find a mapping of the input space into a new feature (vector) space in which problem solving is easier.

Formally, a kernel function is a binary function $K : XxX-> [0, \infty)$ that maps a pair of objects $x, y \in X$ to their similarity score $K(x, y)$. The kernel function must satisfy

$$\forall x, y \in X : k(x, y) = \langle \phi(x), \phi(y) \rangle, \tag{8.3}$$

where $\phi : X \to F \subseteq \mathbb{R}^n$ is a mapping from the input space X to an vector space F. The mapping function ϕ transforms each instance $x \in X$ in a feature vector $\phi(x) = (\phi_1(x), \phi_2(x), \dots, \phi_m(x))$, where $\phi_i : X \to \mathbb{R}$, with no need to know the explicit representation of the x. Then, the mapping function ϕ allows to express $K(x, y)$ as the dot-product of the features vectors of the input objects x and y.

$$\forall x, y \in X : k(x, y) = \langle \phi(x), \phi(y) \rangle = \sum_{i=1}^{m} \phi_i(x).\phi_i(y). \tag{8.4}$$

Giuliano et al. [2006] have developed two different sequence kernels: global context and local context kernels. Furthermore, since several studies [Zhang et al., 2006, Li et al., 2008, Kim et al., 2008, Reichartz et al., 2009] have shown that the combination of kernels overcomes the performance of the individual kernels, they also proposed a linear combination of their global and local kernels. Any kernel function can work with any kernel-based algorithms. Support-vector machines (SVMs) or nearest neighbor classification are examples of machine learning algorithms that can be formulated as kernel-based algorithms. Giuliano et al. [2006] used SVMs, in particular, the SVM package LIBSVM [Chang and Lin, 2001] to embed their proposed kernels and perform the experiments.

8.2.1. Global Context Kernel

The global context kernel defined in [Giuliano et al., 2006] is designed to discover the presence of a relation between two entities. It is based on the following observation in the work of Bunescu and Mooney [2006]: "when a sentence asserts a relationship between two entity mentions, it generally does this using one of the following three patterns: Fore-Between, Between, and Between-After". That is, a relationship between two entities is usually expressed using the words that appear before and between both entities, or just between them, or between and after the two entities.

Example 29 Fore-between, between, and between-after contexts

Fore-Between: tokens before and between the two drugs. For instance: *Interaction between $\langle DRUG \rangle$ and $\langle DRUG \rangle$.*
Between: only tokens between the two drugs. For instance: *$\langle DRUG \rangle$ can interact with $\langle DRUG \rangle$'.*
Between-After: tokens between and after the two drugs. For instance: *$\langle DRUG \rangle$ taken concurrently with $\langle DRUG \rangle$, may affect blood levels'.*

While [Bunescu and Mooney, 2006] represented each context with subsequence of words, POS tas, entity and shallow syntactic types, in [Giuliano et al., 2006] each context is only represented by a bag-of-words. To calculate the similarity between two different global contexts, they propose a n-spectrum kernel [Shawe-Taylor and Cristianini, 2004], which counts common ngrams that two contexts have in common. Formally, given a relation instance R, its global context C is represented as a row vector, as follows:

$$\phi_C(R) = (tf(t_1, C), tf(t_2, C), \dots, tf(t_m, C)) \in \mathbb{R}^m \tag{8.5}$$

where $tf(t_i, C)$ counts the number of occurrences of the token t_i in the context C. The tokens of the entities in C are not taken into account at the calculation of $\phi_C(R)$, however, punctuation and stops words are included. To improve the classification performance, they extended the definition of ϕ_C to embed n-grams of (contiguous) tokens up to n = 3. For each context (fore-between, between, or between-after), a n-gram kernel method is obtained by substituting $\phi_C(R)$ into equation 8.4. The global context

kernel (see equation 8.6) is defined as the sum of the n-grams kernels that work on the fore-between (K_{FB}), between (K_B) and between-after (K_{FBA}) contexts, respectively.

$$K_{GlobalContext}(R_1, R_2) = K_{FB}(R_1, R_2) + K_B(R_1, R_2) + K_{BA}(R_1, R_2) \qquad (8.6)$$

Figure 8.2 and 8.3 show some examples in which the global context kernel is calculated with n-gram=1 and n-gram=2 respectively, to estimate the similarity between relation instances.

$\mathsf{K}_{\mathsf{Global\ context}}$("**DRUG** may interact with **DRUG**", "**DRUG** may interact with **DRUG**, **OTHER**, **OTHER**")=3

$\mathsf{K}_{\mathsf{Global\ context}}$("**DRUG** increases the toxicity of **DRUG**", "**DRUG** may decrease the effect of **DRUG**, **OTHER**, and **OTHER**")=2

$\mathsf{K}_{\mathsf{Global\ context}}$("**DRUG** may interact with **DRUG**", "**DRUG** may decrease the effect of **DRUG**, **OTHER**, and **OTHER**")=1

$\mathsf{K}_{\mathsf{Global\ context}}$("**DRUG** may interact with **DRUG**", "Coadministration of **DRUG** with **DRUG** may increase the risk of toxicity",)=1

$\mathsf{K}_{\mathsf{Global\ context}}$("Coadministration of **DRUG** with **DRUG** may increase the risk of toxicity", "Coadministration of **DRUG** with **DRUG** may increase **OTHER** exposure",)=5

Figure 8.2: Global context kernel (n-gram=1).

$\mathsf{K}_{\mathsf{Global\ context}}$("**DRUG** may interact with **DRUG**", "**DRUG** may interact with **DRUG**, **OTHER**, **OTHER**")=2

$\mathsf{K}_{\mathsf{Global\ context}}$("**DRUG** may interact with **DRUG**", "**DRUG** may decrease the effect of **DRUG**, **OTHER**, and **OTHER**")=0

$\mathsf{K}_{\mathsf{Global\ context}}$("Coadministration of **DRUG** with **DRUG** may increase the risk of toxicity", "Coadministration of **DRUG** with **DRUG** may increase **OTHER** exposure",)=2

Figure 8.3: Global context kernel (n-gram=2).

8.2.2. Local Context Kernel

The local context kernel is based on the hypothesis that the context information of the candidates entities is especially useful in verifying if there is a relationship between them. In particular, windows of limited size around the entities provide useful clues to identify the roles of the entities within a relation. Thus, Giuliano et al. [2006] use the information provided by the two local contexts of the candidate interacting entities, called left and right local context respectively. They consider a context window of $W = \pm2$ tokens around the candidate entity, that is:

$$C = t_{-w}, \ldots, t_{-1}, t_0, t_1, \ldots, t_{+w}, \qquad (8.7)$$

where t_0 is the token of the candidate entity. The following features are proposed to represent each local context:

- *Token*: the token itself.

- *Lemma*: the lemma of the token.

- *PoS*: the PoS tag of the token.

- *Stem*: The stem of the token.

- *Orthographic*: This feature maps each token into equivalence classes that encode features such as capitalization, punctuation, numerals

Each example is basically represented as an instance of the original sentence with the two candidate entities properly annotated, using the tag *DRUG*. The roles of the candidates are labelled with the tags A (agent) and T (target). In our case, agent is always the first argument and target the second argument. Any other entity and tokens that are not entities are labelled O. In our approach, we use *stem* feature instead of *lemma*, because MMTx does not provide lemmatization. To obtain the stems, we use the Porter algorithm [Porter, 1980]. Figure 8.4 shows a sentence and the representation of one of its relation intances. The local and right local contexts for this relation instance are also shown.

LEFT   **RIGHT**

Aspirin may decrease | the | effects of **probenecid,** sulfinpyrazone

and phenylbutazone

CLEFT $\Bigl\{$
Aspirin&Aspirin&noun&Aspirin&**DRUG**&**A**|
may&may&verb&may&O&O|
decrease&decrease&verb&decrease&O&O|
the&the&det&the&O&O|

CRIGHT $\Bigl\{$
effects&effect&noun&effect&O&O|
probenecid&probenecid&noun&probenecid&**DRUG**&**T**|
,&,&comma&,&O&O|
sulfinpyrazone&sulfinpyrazone&noun&sulfinpyrazone&O&O|
and&and&conj&and&O&O|
phenylbutazone&phenylbutazone&noun&phenylbutazone&O&O|

Figure 8.4: Example of left and right local contexts (n-gram=2) in a relation instance.

The local context can be represented as a row vector, as follows:

$$\phi_C(R) = (f_1(C), f_2(C), \ldots, f_m(C)) \in {0, 1}^m \tag{8.8}$$

where f_i is a feature function that return 1 if it is active in the specified position of the context window C, 0 otherwise (see figure 8.5). K_{left} and K_{right} are defined by substituting the embedding of the left and right local context into equation 8.4, respectively. Then, the local context kernel can be defined as the sum of the left context kernel and right context kernel, as follows:

$$K_{LocalContext}(R_1, R_2) = K_{left}(R_1, R_2) + K_{right}(R_1, R_2) \tag{8.9}$$

Figure 8.5 shows the mapping function ϕ for the right local context of the relation instance in figure 8.4. $K_{LocalContext}$ differs substantially from $K_{GlobalContext}$ as it considers the ordering of the tokens and the features space is enriched with PoS tags, lemmas, stems and orthographic features.

Aspirin *may decrease the* effects of **probenecid,** *sulfinpyrazone and phenylbutazone*

$$
\Phi_{RIGHT}(R) =
\left\{
\begin{array}{ll}
\text{Token}_{-2}(\text{effects}) & 1 \\
\text{Lemma}_{-2}(\text{effects}) & 1 \\
\text{PoS}_{-2}(\text{effects}) & 1 \\
\text{Stem}_{-2}(\text{effects}) & 1 \\
\text{Ortho}_{-2}(\text{effects}) & 1 \\
\ldots & \\
\text{Token}_{-1}(\text{of}) & 1 \\
\text{Lemma}_{-1}(\text{of}) & 1 \\
\text{PoS}_{-1}(\text{of}) & 1 \\
\text{Stem}_{-1}(\text{of}) & 1 \\
\text{Ortho}_{-1}(\text{of}) & 1 \\
\ldots & \\
\text{Token}_{1}(\text{.}) & 1 \\
\text{Lemma}_{1}(\text{.}) & 1 \\
\text{PoS}_{1}(\text{.}) & 1 \\
\text{Stem}_{1}(\text{.}) & 1 \\
\text{Ortho}_{1}(\text{.}) & 1 \\
\ldots & \\
\text{Token}_{2}(\text{sulfinpyrazone}) & 1 \\
\text{Lemma}_{2}(\text{sulfinpyrazone}) & 1 \\
\text{PoS}_{2}(\text{sulfinpyrazone}) & 1 \\
\text{Stem}_{2}(\text{sulfinpyrazone}) & 1 \\
\text{Ortho}_{2}(\text{sulfinpyrazone}) & 1 \\
\end{array}
\right.
$$

Figure 8.5: Mapping function ϕ for the right local context (window-size=2) of the relation instance in figure 8.4.

8.2.3. Shallow Linguistic Kernel

To integrate information from heterogeneous feature spaces (such as tokenization, PoS tags, or entity tagging), both kernels must be normalized:

$$
K(x_1, x_2) = \frac{\langle \phi(x_1), \phi(x_2) \rangle}{\parallel \phi(x_1) \parallel \parallel \phi(x_2) \parallel}
\tag{8.10}
$$

where $\phi(.)$ is the embedding vector and $\parallel . \parallel$ is the 2-norm.

Finally, Giuliano et al. [2006] define a shallow linguistic kernel that is a linear combination of the global and local kernels. It is defined as follows:

$$
K_{ShallowLinguistic}(R_1, R_2) = K_{GlobalContext}(R_1, R_2) + K_{LocalContext}(R_1, R_2)
\tag{8.11}
$$

Experiments were performed on the two biomedical data sets: the AImed and LLL corpora. They adopted the evaluation methodology $OAOD$ (one answer per occurrence in the document) [Lavelli et al., 2004], that is, each individual occurrence of a protein interaction must be extracted from the document. Experiments were performed using the correct named entities, that is, those manually annotated in the corpora. The results obtained on the AIme corpus are: precision=60.0%, recall=57.2%, and f-measure=59%. Better performance was achieved on the LLL corpus, with a precision of 62.1%, a recall of 61.3%, and an f-measure of 61.7%.

8.3. Evaluation

Giuliano et al. [2006] have developed the java tool for relation extraction (jSRE) to implement their shallow linguistic kernel. This tool has shown good performance in both general and biological domains [Giuliano et al., 2006, 2007b]. Our objective is to evaluate the shallow linguistic kernel in a domain different from the domains in which it has already been used (news [Giuliano et al., 2007b,a] and PPIs

[Giuliano et al., 2006]). We conducted a set of experiments to study the results obtained in the extraction of DDIs.

The layout of the section is split into two main parts. The first one describes how we have generated examples from the DrugDDI corpus and built the datasets for training and testing. The second part describes the performed experiments and the obtained results.

8.3.1. Datasets

In this approach, DDI extraction is formulated as a supervised learning problem, in particular, as a classification task. Therefore, a crucial task is to generate suitable datasets to train and test a classifier from the DrugDDI corpus.

The DrugDDI corpus consists of 579 files. The average number of sentence per document is 10.3, and the average number of tokens per document is 211.5. The corpus contains a total of 5,806 sentences with at least two drugs and a total of 3,160 drug-drug interactions. Since most of the existing approaches for relation extraction usually assume that the argument entities of the relation occur in the same sentence, we have only considered the interactions between drugs within the same sentence. Although there may be relations between drugs in different sentences, they have not been annotated in the DrugDDI corpus . The average number of interactions per document is 5.46 and per sentence 0.54.
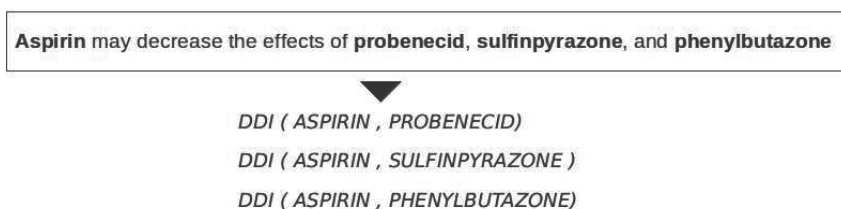

Figure 8.6: Real DDIs occurring in a sentence.

Figure 8.7: Examples generated from the sentence in figure 8.6.

1) DRUG may decrease the effects of **DRUG**, **OTHER**, and **OTHER**   => label = 1

2) DRUG may decrease the effects of **OTHER**, **DRUG**, and **OTHER**   => label = 1

3) DRUG may decrease the effects of **OTHER**, **OTHER**, and **DRUG**   => label = 1

4) OTHER may decrease the effects of **DRUG**, **DRUG**, and **OTHER**   => label = 0

5) OTHER may decrease the effects of **DRUG**, **OTHER**, and **DRUG**   => label = 0

6) OTHER may decrease the effects of **OTHER**, **DRUG**, and **DRUG**   => label = 0

Figure 8.8: Labeling candidate drugs

The simplest way to generate examples to train a classifier for a specific relation R is to enumerate all possible ordered pairs of entities in sentences. We have proceeded in a similar way. Given a sentence S, as the own shown in figure 8.6, with at least two drugs, we can define D as the set of drugs in S. The set of generated examples for this sentence S, is defined as follows:

$$\{(D_i, D_j) : D_i, D_j \epsilon D, 1 <= i, j <= N, i \neq j, i < j\} \qquad (8.12)$$

The works presented in [Giuliano et al., 2006] and [Giuliano et al., 2007b] detect the presence of a given relation and also identify the roles in the relation for each of the entities. So each example is the copy of the original sentence S in which the candidates are assigned distinctive attributes to specify their roles ($AGENT, TARGET$) in the relation. In our case, although there are asymmetric interactions between drugs, the roles of the interacting drugs have been neither included in the annotation of the corpus, nor addressed in this thesis. Hence we enumerate the candidate pairs without taking into account the drugs order, that is, (D_i, D_j) and (D_j, D_i) are considered as one only candidate pair, that is,

$$(D_i, D_j) = (D_j, D_i) \qquad (8.13)$$

Since we do not take into account the order of the drugs in the sentence, each example is the copy of the original sentence S in which the candidates are assigned the tag $DRUG$, while the drugs not involved are assigned the tag $OTHER$ (see figure 8.8). The set of possible candidate pairs is the set of 2-combinations of the whole set of drugs in the sentence S, and thereby, the number of examples is

$$C_{N,2} = \binom{N}{2} \qquad (8.14)$$

where N is the number of drugs in S. If the interaction exits between the two candidate drugs, then the example is labeled 1, otherwise, it is labeled 0. The sentence shown in figure 8.6 contains four drugs (*aspirin*, *probenecid*, *sulfinpyrazone* and *phenylbutazone*), and thereby, the total number of examples generated is $C_{4,2} = \binom{4}{2} = 6$. Figure 8.7 shows the examples generated from this sentence. In each example, the two drugs selected are considered as candidate interacting drugs, while the other drugs are not considered as they do not participate in the interaction. It is not necessary to generate examples for each ordered pair of drugs since this approach considers the drug-drug interactions as symmetric relations.

Once we have generated the set of relation instances from the DrugDDI corpus, we have split this set in order to build the datasets for training and testing the different models. In this work, the extraction problem is reformulated as a binary classification task. The relation extraction is performed in two distinct phases: *learning* and *classifying* (see figure 8.9). First, one model must be learned for drug-drug interactions from training dataset. This model is then applied in order to identify the interactions occurring in the documents that form the testing dataset.

Table 8.1 shows the total number of examples generated from the DrugDDI corpus. In our corpus, sentences that contain at least two drugs are selected obtaining 3,775 sentences with 3,313 drugs. Possible pairs of drugs are 30,757, specifically, 3,160 are DDIs, and 27,597 are not. The corpus DrugDDI consists of 579 files, we randomly choose 75% of them to build the training dataset, which will be used to train and adapt the different models. The remaining 142 files are used in the final evaluation of the best model. Table 8.2 shows the distribution of the documents, sentences, drugs and DDIs in each dataset. So, the
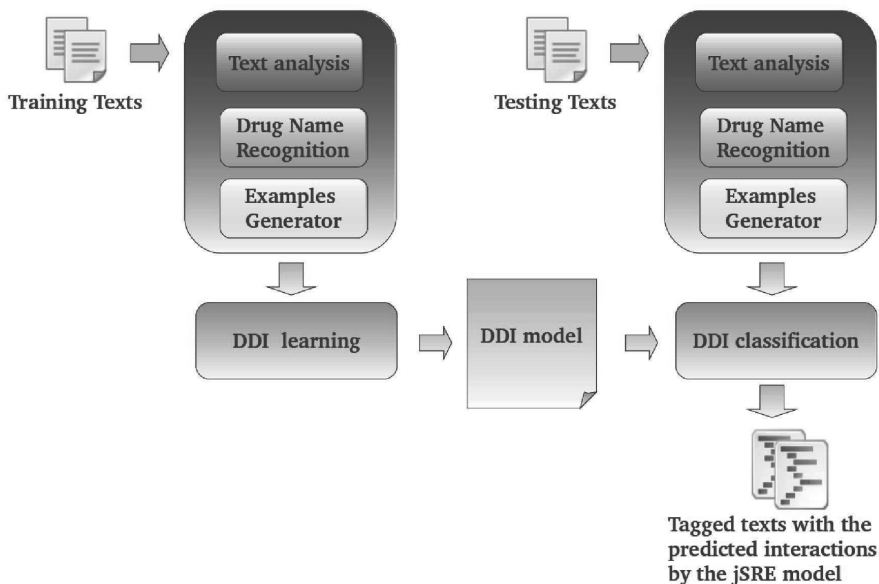
Figure 8.9: Architecture for DDI extraction

Positives	Negatives	total
3,160 (10.27%)	27,597 (89.72%)	30,757

Table 8.1: Total of positive and negative examples (relation instances) generated from the DrugDDI corpus.

dataset for training and tuning consists of 437 files and contains 4,578 sentences, 2,560 drugs and 2,421 DDIs. This dataset will be used to build several models based on different configurations of the jSRE tool. The final testing dataset consists of 142 files, containing 1,228 sentences, 753 drugs and 739 DDIs.

Set	Documents	Sentences	Drugs	DDI
Training	437	4,578	2,560	2,421
Final Test	142	1,228	753	739
Total	579	5,806	3,313	3,160

Table 8.2: Training and testing datasets.

Table 8.3 shows the distribution of the positive and negative examples in the different datasets. Around 90% of instances in the training dataset are negative examples, and only almost a 10% are positive examples. The distribution between positive and negative examples is similar in the final testing dataset.

Set	Documents	Examples	Positives	Negatives
Training	437	25,209	2,421 (9,6%)	22,788 (90,4%)
Final Testing	142 (25%)	5,548	739 (13,3%)	4,809 (86,7%)
Total	579	30,757	3,160 (10,27%)	27,597 (89.73%)

Table 8.3: Distribution of the positive and negative examples in the training and testing datasets.

8.3.2. Experimental results

Since one of our main objectives is to study the influence of the configuration parameters of the jSRE tool (window size of the local context and n-gram of the global context) on the final performance, we have designed a set of experiments in which these parameters are varied.

We consider as baseline system, so called *allDDIs*, the case in which every relation instance is classified as a DDI, that is, as a positive example. This baseline yields the maximum recall, but a low precision. The baseline system is evaluated on final testing dataset achieving a baseline precision of 11% and a baseline f-measure of 19% (see table 8.4, row 1). We used 10-fold cross validation to train the classifier on the training dataset. For each run, nine folds are used to train a model that is evaluated on the other fold. The folds were built considering that examples from the same sentence must belong to the same fold. We follow the evaluation methodology *OAOD* [Lavelli et al., 2004], that is, each individual occurrence of a DDI must be extracted from the document.

We set out various experiments varying the values of the configuration parameters concerning the local (window-size) and global contexts (n-grams) in order to obtain the best performance. Giuliano et al. [2006] only considered n-grams up to n=3 (global context) and only reported results with window size ±2 (local context).

Experiment	P	R	F	Time (ms.)
allDDIs	0.11	1	0.19	1,534
n-gram=1, window-size=1	0.51	0.78	0.62	3,915,762
n-gram=2, window-size=1	0.54	0.75	0.63	6,039,573
n-gram=3, window-size=1	**0.57**	0.72	**0.64**	8,353,649
n-gram=4, window-size=1	0.56	0.73	0.63	8,075,174
n-gram=5, window-size=1	0.56	0.72	0.63	7,930,885
n-gram=1, window-size=2	0.55	0.76	**0.64**	9,291,000
n-gram=2, window-size=2	0.55	0.73	0.63	6,976,556
n-gram=3, window-size=2	0.55	0.72	0.62	9,705,327
n-gram=4, window-size=2	0.53	0.76	0.63	9,286,145
n-gram=5, window-size=2	0.56	0.72	0.63	9,304,274
n-gram=1, window-size=3	0.52	0.77	0.62	5,971,677
n-gram=2, window-size=3	0.56	0.75	**0.64**	8,165,894
n-gram=3, window-size=3	0.55	0.75	**0.64**	10,224,599
n-gram=4, window-size=3	**0.57**	0.72	**0.64**	10,433,904
n-gram=5, window-size=3	**0.57**	0.73	**0.64**	14,945,000
n-gram=1, window-size=4	0.52	0.76	0.62	6,874,866
n-gram=2, window-size=4	0.55	0.72	0.62	9,022,820
n-gram=3, window-size=4	0.55	0.70	0.62	11,682,749
n-gram=4, window-size=4	**0.57**	0.71	0.63	11,742,036
n-gram=5, window-size=4	0.56	0.70	0.62	12,163,292
n-gram=1, window-size=5	0.52	**0.80**	0.63	8,201,097
n-gram=2, window-size=5	0.56	0.73	0.63	10,478,464
n-gram=3,window-size=5	**0.57**	0.70	0.63	12,078,133
n-gram=4, window-size=5	0.55	0.70	0.62	12,732,470
n-gram=5, window-size=5	**0.57**	0.69	0.63	12,578,524

Table 8.4: Experiment results based on different configurations of jSRE tool

Experiment results (see table 8.4) show that the performance does not differ significantly from one configuration to another. Precision ranges from 51% to 57%, recall from 72% to 80%. The highest precision (57%) is achieved when n-gram is equal to 3 and window-size is equal to 1 (72% recall). Other configurations also achieve this precision, however, they yield lower recall than 72% and take much more time to train its model. The highest recall is achieved when n-gram is equal to 1 and window-size is equal to 5 (52% precision). Increasing the value of the parameter n-gram does not influence the results since it is few probable that two relation instances share long n-grams. Similarly, the choice of the parameter window-size does not seem to affect the performance significantly. This is logic because this parameter is designed to identify the roles of the entities within a relation, which are not tackled in these experiments. Table 8.4 also shows the time needed to train each model. While the increase of the configuration parameters does not seem to improve the results, however results in a drastic increase of the training time. Among all trained models, we choose one that minimizes the training time and maximizes

the precision (n-gram=3, window-size=1), because it avoids overloading end users with too many false positives in the extraction of DDIs.

In addition, we have evaluated each kernel separately, in order, to analyze the contributions of the global and local kernels to the overall shallow linguistic kernel. Although, several global and local kernels were trained with different values of their parameters (n-gram and window-size respectively), table 8.5 only presents the best results corresponding to the best configurations. Results show that global context is more useful than the local one in detecting DDIs. Giuliano et al. [2006] used the local context kernel to identify the roles of the candidate entities within a relation. Although in our approach the roles are not tackled, the results show that the local context kernel also assists in the detection of DDIs since the combination of both kernels achieves to improve the performance, especially the precision.

Kernel	P	R	F	Time (ms.)
Global Context (n-gram=3)	0.53	0.72	0.61	6,731,856
Local Context (window-size=2)	0.51	0.68	0.58	3,453,999
Shallow (n-gram=3, window-size=1)	0.57	0.72	0.64	8,353,649

Table 8.5: Comparative analysis of global, local and shallow kernels

Finally, the shallow kernel (trained with n-gram=3 and window-size=1) was evaluated on the final testing dataset achieving a precision of 50%, a recall of 67% and an f-measure of 57% (see table 8.6). It is an improvement of almost 40% with regard to the baseline f-measure. The results are lower than those reported above for its cross validation experiment (table 8.4, row 4). This may be due to the final testing dataset can contain examples more complicated than those in the folds generated from the training dataset. To validate this hypothesis, we have evaluated some of the above models on the final testing dataset. Every model shows lower performance than in cross validation.

	P	R	F
allDDIs	0.11	1	0.19
n-gram=1, window-size=2	0.45	0.63	0.53
n-gram=2, window-size=3	0.49	0.69	0.57
n-gram=2, window-size=2	0.49	0.71	0.58
n-gram=3, window-size=1	0.50	0.67	0.57
n-gram=3, window-size=3	0.50	0.67	0.57
n-gram=4, window-size=3	0.50	0.67	0.57

Table 8.6: Final results obtained by the shallow kernels.

Learning curves are useful to show the results achieved by the learning process for different training sizes. We now describe the process of construction of the learning curves. We must estimate the f-mesure, precision and recall for different training sizes. We have used the configuration of the jSRE that has shown the best results in the previous experiments (window-size=1, n-gram=3). In order to generate datasets of different sizes, the training dataset is split into 10 folds of approximately the same size. Each fold consists of 10% of the total instances, that is, each fold roughly contains 3,076 instances. We impose the following restriction: relation instances of the same sentence must always be in the same fold. We can merge the folds in order to generate new sets of different sizes. So, the first size is 3,076, and this is successively increased adding a new fold: the second one is 6,152, the third one is 9,228, and so on until the total number of instances: 30,757. We may randomly combine all folds in order to obtain all possible sets for each one of the training sizes. For example, building a set using the 30% of instances (92,289) requires to merge three different folds, and for this case, we may generate $\binom{10}{3} = 120$ different sets. For the training size 6,152, we may generate $\binom{10}{2} = 45$ different sets. Training each model and evaluating on the final dataset, it a very time consuming task. Therefore, we have decided to generate ten sets from all possible combinations for each training size. Finally, a model is learned on each set and evaluated on the final dataset. In order to obtain the results for a given size, we calculate the average of the f-measure, precision and recall of the models that have been trained on the ten sets of that size. Table 8.7 shows the performance obtained with each of the training sizes.

Figure 8.10 shows the learning curves generated. We can see that increasing the training size, the performance is hardly improved. The annotation of new sentences do not seem to help for the improvement of the performance (from 40%). The dashed lines represent the precision and f-measure obtained by the baseline experiment (*allDDIs*). It is obvious that the baseline achieves a recall of 100%, but only

size	Avg. P	Avg. R	Avg. F
3,075.7 (10%)	0,45	0,53	0,48
6,151.4 (20%)	0,47	0,6	0,53
9,227.1 (30%)	0,47	0,64	0,55
12,302.8 (40%)	0,48	0,66	0,56
15,378.5 (50%)	0,49	0,67	0,57
18,454.2 (60%)	0,49	0,68	0,57
21,529.9 (70%)	0,5	0,68	0,58
24,605.6 (80%)	0,49	0,68	0,57
27,681.3 (90%)	0,49	0,69	0,57
30,757 (100%)	0,48	0,7	0,57

Table 8.7: Average precision, recall and f-measure

a precision equal to the percent of positive examples in the final test, that is, a 12% of precision, and an f-measure of 22%.
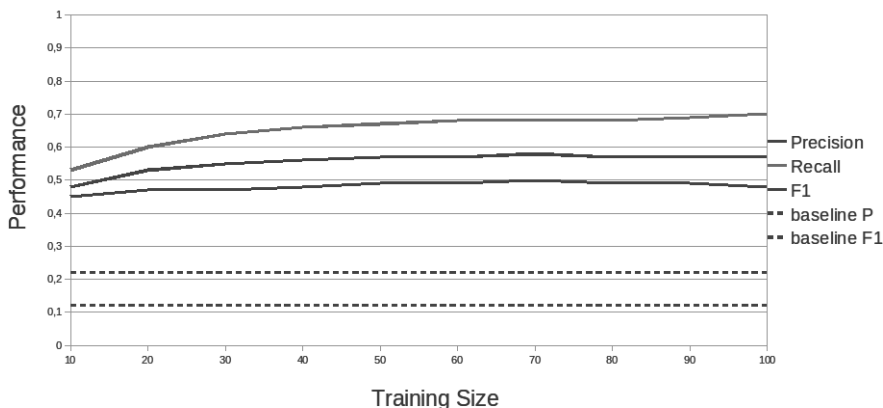

Figure 8.10: Learning Curves.

As it was seen in the previous subsection, from all pair of drugs occurring in our corpus (30,757), only 10% of them (3,160) are drugs that interact. That is, only a 10% of relation instances are DDIs (positive examples). For this reason, we want to study the impact of our imbalanced dataset on the performance of the kernel-based method. A common problem in most of the machine learning algorithms is their inability to accurately learn from imbalanced data. The rules learnt by machine learning algorithms to describe the minority classes are fewer and weaker than those of the majority classes. This is due to minority classes are often underrepresented [He and Garcia, 2009, Van Hulse and Khoshgoftaar, 2009]. There are different solutions for imbalanced learning such as sampling, cost-sensitive, kernel and active learning methods. A detail description of these solutions can be found in [He and Garcia, 2009]. Regarding the sampling methods, they modify the imbalanced dataset by some mechanisms in order to provide balanced distribution. The two most common mechanisms are *oversampling* and *undersampling*. Basically, *oversampling* adds copies of the examples from the minority classes, while *undersampling* consists of removing examples from the majority classes. The main advantage is that these mechanisms are simple to implement, however, they also present some drawbacks. *Undersampling* involves a considerable information loss, in which discriminative features to differentiate among classes may be discarded, adversely affecting the classification performance [Van Hulse et al., 2009]. In this case, the ideal would be to eliminate redundant examples and that are very close to those of the minority classes. Regarding they *oversampling* technique, replicating existing minority class examples may not add any actual information to the dataset. Moreover, adding examples does not only lead to a significant increase in computational costs, but also overfitting is very likely to occur [Seiffert et al., 2009]. We believe that the computational time is crucial for clinical applications like the automatic extraction of DDIs from

texts. The models shown in table 8.4 already require high training times. Therefore, we have decided to apply the *undersampling* technique in order to reduce the computational costs and avoid the overfitting problem. Thus, we have performed three different experiments with imbalanced data and balanced data, which are described below:

1. *Imbalanced*: both training and testing dataset are unbalanced. This experiment is the most nearest to the real situation. Our textual corpus has been collected from a pharmacological database (DrugBank), specifically, from a text field that describes DDIs for a given drug. It is foreseeable that this number will be very much less if we build a corpus from abstracts from Medline (although we will build the corpus using keywords that express DDIs) since many of these abstracts do not contain necessary information about DDIs.

2. *TrainingBalanced:* we used the *undersampling* mechanism to randomly remove negative examples from the training dataset, while we maintained unbalanced the testing dataset. After a model is obtained from the balanced training dataset, then the trained model is applied to the unbalanced testing dataset. Here, our objective is to analyze if the model learned from the balanced dataset is useful with real imbalanced data. The drawback of this experiment is that the size of training dataset is reduced notably, for equilibrating the number of both types of examples. Our hypothesis is that if the amount of positive and negative examples is the same in the training set, the model can distinguish better the minority class (DDIs).

3. (*Balanced*): we also applied the *undersampling* mechanism to randomly reduce both training dataset and testing dataset to obtain the same amount of positive and negative examples in each dataset. In spite of this is an unreal situation, we also tackled it in order to evaluate the performance of the kernels in balanced training and testing data.

Each experiment is compared to the baseline *allDDIs*, in which all examples are labeled as DDIs, that is, as 1. This baseline allows us to estimate the improvement achieved in each experiment above described. Table 8.8 shows the results obtained in each experiment. In the first experiment (*Imbalanced*), the baseline only achieves a 11% of precision (it is obvious since the proportion of examples positives is a 11%). The learned model achieves a good performance with a 66% of f-measure. This is an improvement of 55% respect to the baseline f-measure. The sencond experiment, *TrainingBalanced*, obtains high recall (91%), however, the precision is quite low (36%). Therefore, balancing training data helps to improve the recall but the precision is severely degraded. We have also studied the increment in f-measure obtained by each experiment with respect to the baseline f-measure. The increment can defined as follows:

$$Inc(F_{baseline}, F_{jSRE}) = \frac{F_{jSRE} - F_{baseline}}{F_{baseline}} \tag{8.15}$$

Thus, the increment in f-measure with respect to the baseline f-measure (19%) is lower in the *TrainingBalanced* experiment than in the *Imbalanced* experiment. Therefore, we can conclude that balancing the training dataset does not show increased performance. Regarding the last experiment (*Balanced*), its model achieves a good performance. For this experiment, we can conclude that balancing positive and negative examples by undersampling mechanism achieves a better results than those obtained from the imbalanced data. However, the increase over the baseline f-measure is higher in the above experiments performed on imbalanced dataset.

Experiment	P	R	F1	Inc
allDDIs	0.50	1	0.67	–
Balanced	0.82	0.91	0.86	0.25
allDDIs	0.11	1	0.19	–
Imbalanced	0.55	0.82	0.66	2.47
TrainingBalanced	0.36	0.91	0.52	1.74

Table 8.8: Experiment results on imbalanced and balanced datasets.

Regarding the task of classification if a given example is an interaction or not, table 8.9 shows the performance of the experiments for each class (1 DDI, 0 otherwise). The *Balanced* experiment shows good and similar performance for both classes. The other two experiments, *Imbalanced* and *TrainingBalanced*, show good performance in the classification of the negative examples. In the *Imbalanced* experiment, this

may be due to that the amount of negative examples is considerably greater than the number of positive examples, providing strong clues to describe the majority class (non-interactions). However, we must note that in the *TrainingBalanced* experiment, in which the number of negative examples was reduced to equal the number of positive ones in the training dataset, its results for the non-interaction class (that is, negative examples) are still considerably high. Therefore, we believe that it is easier to determine a non-interaction than an interaction.

Experiment	class	P	R	F1
Balanced	0	0.91	0.82	0.86
	1	0.82	0.91	0.86
Imbalanced	0	0.98	0.93	0.95
	1	0.55	0.82	0.66
TrainingBalanced	0	0.99	0.82	0.90
	1	0.36	0.91	0.52

Table 8.9: Experiment results on imbalanced and balanced datasets grouped by class

8.4.   Conclusions

Our major objective was to evaluate the performance of the shallow linguistic kernel-method introduced in [Giuliano et al., 2006] to extract DDIs from biomedical texts. Several experiments have been conducted on the DrugDDI corpus. In particular, we have varied the configuration parameters window-size (local context kernel) and n-grams (global context kernel). Experiments show that the performance does not differ significantly from one configuration to another. Increasing the value of the parameter n-gram does not influence the results since it is few probable that two relation instances share long n-grams. Similarly, the choice of the parameter window-size does not seem to affect the performance significantly. This is logic since we do not distinguish the roles of the interacting drugs, and this parameter is designed to identify them. Among all learned models we decided to choose one that maximizes the precision and the f-measure but minimizes the training time, (window-size=1, n-gram=3). This model is tested on the final testing dataset achieving a precision of 50%, a recall of 67% and an f-measure of 57%.

We are aware that the previous results are not directly comparable with those obtained by our pattern-based approach (described in the previous chapter 7) since it was evaluated on the whole DrugDDI corpus. In order to compare both approaches, the kernel-based approach has been tested on the whole DrugDDI corpus using 10-fold cross validation. Table 8.10 compares the two approximations proposed in this thesis for the extraction of DDIs. The Kernel-based approximation has remarkably overcome the pattern matching method. The more significant improvement is achieved in terms of recall and f-measure. Recall increased up to 82% and f-measure to 66%. This is an increase of almost 57% in recall rates, and almost 33% in f-measure. A minor improvement is also achieved for the precision, with an increase of 6.31%. Therefore, we can conclude that machine learning-approach is more efficient than the pattern-based approach to tackle the extraction of drug-drug interactions from texts.

Approach	P(%)	R(%)	$F_{\beta} = 1(\%)$
Pattern-based approach	48.7	25.7	33.6
Kernel-based approach	55	82	66

Table 8.10: Experiment results: patterns vs kernels.

We cannot compare this work with any other approach because we are the first who have addressed the DDI extraction. Our experiments have been performed on a specific corpus for our task, and different to the corpora used in Giuliano et al. [2006]. Giuliano et al. [2006] performed several experiments on two different biomedical corpora: AImed and LLL. Their experiments are performed using both the correct named entities, that is, those manually annotated in the corpora. The results obtained on the AImed corpus show a precision of 60.0%, a recall of 57.2%, and f-measure of 59%. Better performance is achieved on the LLL corpus, with a precision of 62.1%, a recall of 61.3%, and an f-measure of 61.7%. A possible explanation of our results being lower than those, could be that our performance shows a remarkable impact of automatic entity recognition on the relation extraction task. It is predictable that if drug names were manually labeled in our corpus, our results will significantly improve. On the other hand,

the LLL corpus is smaller than the DrugDDI corpus, however, its average number of interaction per sentence is higher than that in the DrugDDI corpus (see table 8.11). We believe that a high density of interactions could positively influence on the performance since sentences in the LLL corpus are focused on the description of interactions, while sentences in our DrugDDI corpus may be less discriminative.

Corpora	Dataset	Sentences	interactions	Avg.
LLL	train	80	271	3.4
	test	82	166	2
DrugDDI	train	4,578	2,421	0.5
	test	1,228	739	0.6

Table 8.11: Comparation between LLL and DrugDDI corpora.

The DrugDDI corpus shows a great imbalance distribution between positive and negative examples because only 10% of relations instances are drug-drug interactions. We have conducted thee experiments to study the influence of the imbalanced dataset on the performance of the kernel-method. Experiments evaluated on imbalanced data show that balancing the training dataset does not show increased performance. The experiment using balanced training and testing dataset achieves better results than those obtained from the imbalanced dataset. However, the increase over the baseline performance is higher in imbalanced dataset.

To conclude, we believe that the good performance achieved using the shallow linguistic kernel provide a higher baseline, being possible to measure improvements using other methods that use full syntactic or semantic information. We propose several specific ideas for future work:

- Evaluate each of the three contexts (before, between and after) separately, assigning different weights to them. Here, our objective will be to study which of the three contexts is the most discriminative for DDIs.

- Evaluate the performance of the kernel-method when the drug names are manually annotated.

- Label the roles of drugs in the DrugDDI corpus in order to evaluate the contribution of the local kernel in their detection.

- Define a semantic kernel that uses semantic information such as semantic type from UMLS or drug families obtained by the DrugNer module.

- Design parse tree or dependency graph kernels for the extraction of DDIs.

- Evaluate other solutions for imbalanced learning such as hybrid sampling or cost-sensitive methods.

Chapter 9

Conclusions

9.1. Conclusions and Future Work

A DDI occurs when one drug influences the level or activity of another drug. DDIs are common adverse drug reactions (ADR), which can lead to significant morbidity and mortality. In addition, DDIs are a direct cause of the increase of health care costs because they account for 16.6% of adverse drug reactions causing hospitalization [Pirmohamed et al., 2004].

There are several databases supporting health care professionals in the detection of DDIs, however they are rarely complete. A great deal of the most current and valuable information on DDIs is unstructured and hidden in articles, scientific journals, books and technical reports. Therefore, the medical literature is probably by far the most effective system for detection of DDIs [Aronson, 2007]. The great amount of drug interaction databases and the deluge of published research have overwhelmed most health care professionals because it is not possible to be kept up-to-date of everything published about drug-drug interactions. Information Extraction from unstructured data sources can be of great benefit providing an interesting way to reduce the time spent by health care professionals on reviewing the literature. Moreover, the development of tools for the automatic extraction of drug-drug interactions is essential for improving and updating the drug knowledge resources. The major goal of this thesis is to develop and improve IE techniques applied to biomedical texts, in particular, in the scenario of DDIs. Next, we discuss whether the objectives proposed in this thesis (see section 1.2) have been achieved.

Our first objective is the creation of an annotated corpus of DDIs. The review presented in section 3.1 showed that most biomedical annotated corpora for relation extraction are focused on the biological domain and there is no corpus for DDIs. We have built the first annotated corpus of DDIs, the DrugDDI corpus, which has allowed us to automatically evaluate the different approximations proposed in this thesis to extract DDIs. To the best of our knowledge, the problem of producing an annotated corpus for DDI extraction has not been explored in the depth and extent reported in this chapter. Also the resulting corpus is the most semantically rich annotated resource for pharmacological text processing built up to date. The DrugDDI corpus consists of 579 documents from the DrugBank database and a total of 9,601 sentences. The documents were semantically and syntactically analyzed by MMTx and annotated with linguistic information including sentence boundaries, tokenization, phrase boundaries and phrase semantic classification provided by MMTx. MMTx integrates several biomedical knowledge resources contained within the UMLS system, providing a broad coverage for a huge amount of biomedical terms (objective 3). The corpus contains a total of 3,160 DDIs manually annotated at sentence level with the assistance of an expert pharmacist. DrugDDI is larger than other biomedical corpora, which never exceed 1,100 sentences and 2,662 relationships [Pyysalo et al., 2007]. We now outline directions for future improvements of the DrugDDI corpus:

1. Increase the size of the corpus, using other textual sources such as the bibliographics databases MedLine and EMBASE.

2. Annotate the interactions at document level for capturing those interactions spanning several sentences.

3. Annotate related information on drug-drug interactions such as level of severity, mechanism of action, degree of certainty, drug dosages, time interval between administration of the drugs, etc. They are relevant features to characterize the DDIs and assign them a real clinical significance.

4. From a syntactic point of view, the inclusion of full syntactic information provided by biomedical parsers such as Genia dependency parser [Sagae, 2007] or the one proposed in [Lease and Charniak, 2005] will be considered. In addition, we will manually review the linguistic and semantic analysis provided by the MMTx tool, dealing with the failures of MMTx. We will also annotate negation and modality because they can significantly alter or even reverse the meaning of the sentence.

5. Provide comprehensive annotation guidelines useful for other annotators to cover aspects such as the annotation of interactions involving anaphoric and cataphoric expressions or the treatment of uncertain interactions. This guideline is crucial in order to achieve high quality annotations.

6. Annotate the corpus by various pharmacists and measure the inter-annotator agreement. This will allow us to test the quality of the annotation itself and check if our guidelines are robust, consistent and easily identifiable.

The availability of the DrugDDI corpus is an important contribution for the development of other approaches for DDI extraction since it provides gold-standard data for evaluation. We believe that this shared corpus should result in an increased focus and rapid advances in the field. Thus, we hope to encourage many researchers to make use of DrugDDI corpus for their research, and expect much feedback from them that would be the most valuable source for further improvement of the corpus.

Our second objective pursued to study the main approaches for biomedical IE. We have reviewed the main techniques used for named entity recognition, anaphora resolution and relation extraction in the biomedical domain. Shortcomings of these techniques in view of our work are identified and some solutions are proposed.

Detecting and classifying drug names occurring in biomedical texts have also been carried out in this thesis (objectives 3 and 4). We have proposed a hybrid method that combines the use of the MMTx tool and a set of affixes recommended by the WHOINN Program to identify and classify drug names. MMTx is an effective program for the automatic processing of the biomedical texts, however, it is not able to provide complete and useful information about pharmacological substances. The affixes allow to recognize drugs not detected by MMTx, and establish suitable information such as their pharmacological families. Although evaluation reveals that affixes alone are not feasible enough in detecting drugs they help to improve the coverage. Combining MMTx and the affixes achieves a precision of 99.1% and a recall of almost 100%. In our experiment, we assumed that the drug name recognition provided by MMTx was correct, because our main objective was not to evaluate the performance of MMTx, but to study if the affixes could help to identify new drugs not detected by MMTx.

In future work, we are planning to provide a more realistic evaluation taking into account the mistakes made by MMTx. It is foreseeable that the performance will be negatively affected. Regarding the drug name classification task, the affixes are able to correctly classify 74.9% of drugs occurring in the texts. The ATC system provides a global standard for classifying medical substances and serves as a tool for drug research, for this reason, we are planning to develop machine learning-based techniques to predict the ATC class for unclassified drugs. Moreover, linking the affixes with the groups of the ATC classification system is an important challenge to be met in future work. Resolving drug acronyms, extending the set of affixes, including additional clues for those affixes that are too short and ambiguous to correctly detect drugs are some challenges for our future research to improve the recall and the precision of drug name recognition and classification tasks.

We have also developed a pipeline prototype to provide a framework (objective 5), allowing to easily combine and evaluate several techniques for coping with the different processes involved in the extraction of DDIs.

The extraction of DDIs requires the resolution of anaphoric expressions in pharmacological texts (objective 6). We have developed two different approaches to address the problem of co-referring expressions in pharmacological literature. In addition, we have built and annotated a corpus in order to analyze the phenomena and evaluate both approaches. To the best of our knowledge, this is the first work that addresses this issue. The first approach is based on a scoring system and obtains results that are similar to those of other systems referred to in the biomedical domain [Castano et al., 2002, Lin et al., 2004, Kim and Park, 2004a, Liang and Lin, 2005]. It shares with these works the use of a set of linguistic features and a semantic resource such as UMLS.

The second approach for anaphora resolution uses Centering Theory [Grosz et al., 1995] in order to select the scope of the anaphoric expressions and assign the correct antecedent. In contrast, a simpler heuristic that selects the closer nominal phrase has been experimentally shown as a useful rule to solve relative pronouns and possessive nominal anaphoras. A key component of both approaches is the use of the biomedical parser MMTx. Unfortunately, this tool only provides shallow syntactic information, so it can be expected that full syntactic parsing improves the performance of the linguistic rules-based method. The scoring-based approach achieves a precision of 77% and a recall of 62%. The linguistic rules-based approach overcomes the scoring approach, obtaining a precision of 84% and a recall of 70%. MMTx makes several syntactic and semantic parsing failures [Divita et al., 2004], which influence negatively on the performance of both approaches.

Neither of the proposed approaches for drug anaphora resolutions have been integrated in the extraction of DDIs. Thus, this objective is still an unsolved goal. Future work will consider the overall contribution of the anaphora resolution approaches to the broader task of DDI extraction. To automatically evaluate the overall contribution, it is necessary to annotate the DrugDDI corpus at document level. Although sources providing information on interactions such as Medline abstracts and DrugBank may share a common literary style, the distribution of interactions is very different and it also deserves investigation. On the other hand, semantic information about drug families provided by the DrugNer process can be valuable for the improvement in the resolution of certain nominal anaphoras. Additional improvements include extending the coverage of the approach to other kinds of biomedical entities, increasing of the size of the corpus in order to make more reliable conclusions, and the application of machine learning techniques, which have been successfully applied in other domains.

In this thesis, we have developed two different approximations for the extraction of DDIs from biomedical texts. The first approximation proposes a hybrid approach that combines shallow parsing and pattern matching to extract relations between drugs from texts (objective 7). Appositions and coordinate structures are detected using shallow syntactic and semantic information provided by MMTx. Complex and compound sentences are broken down into clauses from which relations are extracted by a pattern matching algorithm. The patterns were manually defined by our pharmacist based on her professional experience and the observation of the training corpus. The approach was evaluated on the DrugDDI corpus (annotated dataset) showing a low performance (f-measure of 33.64%). Regarding the resolution of the linguistic constructions, most of the errors are due to mistakes introduced in the MMTx level and the difficulty of resolving nested clauses, so frequent in biomedical texts. On the other hand, the variability of natural language expression makes it difficult for this approach to accurately detect all semantic relations occurring in text since sentences conveying the same relation may be composed lexically and syntactically differently. Inversely, sentences which are lexically common may not necessarily convey the same relation. Thus, our patterns are not enough to identify many of the interactions.

We now point out potential future directions for research on the hybrid approach. The resolution of the different failures of MMTx hindering the successful detection of the linguistic constructions will be addressed. We will also cope with the kinds of appositions not addressed in this work. The improvement of the clause splitting process using machine learning-based techniques as well as the definition of new simplification rules to generate simple sentences from clauses also deserve investigation.

We will carry out a more exhaustive treatment of negation and modality, since these phenomena can significantly alter or even reverse the meaning of a sentence. This approach firstly recognizes the coordinate structures, followed by the appositions, and finally, the clause splitting process is conducted. We will evaluate how the order in which the different constructions are detected affects the performance of the approach. In addition, we will manually annotate the corpus used for the evaluation of the linguistic constructions. On the other hand, the utility of the other corpora such as Genia-GR or Penn Treebank for the evaluation of the syntactic resolution will be studied.

Finally, we will propose the use of the SPINDEL [de Pablo-Sánchez and Martínez, 2009] system to semi-automatically acquire linguistic patterns for DDI extraction. This system was developed by

the Group of Advanced Databases (LABDA) [1] with the aim of semi-automatic generation of valuable resources to build a named entity recognition system for domains and languages with scarse resources. The system bootstraps the acquisition of large dictionaries of entity types and pattern types from a few seeds and a large unannotated corpora, providing good performance for a weakly supervised method.

Our second approximation is based on the kernel-based approach presented by Giuliano et al. [2006] (objective 8). The approach combines two kernel methods, called global context and local context kernels, which integrate the information of the whole sentence where the relation occurs and the context information of the interacting entities, respectively. In particular, we have used their java tool for relation extraction (jSRE), which has been applied to relation extraction with good results in both general and biological domains. jSRE only needs shallow syntactic information such as sentence splitting, tokenization, part-of-speech tagging and lemmatization to build the structured representation of each relation instance.

Several experiments were performed on the DrugDDI corpus, which was split into two sets: training dataset to learn models, and the final testing dataset to evaluate the best models. In the experiments, we have varied the configuration parameters of global (n-gram) and local kernels (window-size). Experiment results show that the performance does not differ significantly from one configuration to another. Therefore, we selected the model that maximized the results and minimized the training time (n-gram=3, window-size=1). This model was evaluated on the final testing dataset, achieving a precision of 51%, a recall of 67% and an f-measure of 58%. These results are not directly comparable with those obtained by our pattern-based approach, since it was evaluated on the whole DrugDDI corpus. For this reason, the kernel-based approach was evaluated on the whole corpus using 10-fold cross validation, overcoming the pattern-based method remarkably (objective 9). Precision was increased from 48,69% to 55%, recall from 25,70% to 82%, and f-measure from 33,64 to 66%. The most significant improvement is achieved in terms of recall and f-measure, with an increase of almost 57% in recall rates, and almost 33% in f-measure.

We cannot compare with any other approach because we are the first who have addressed the DDI extraction problem. Although we have used the same evaluation methodology ($OAOD$) adopted by Giuliano et al. [2006] (this work tackled the extraction of PPIs), our experiments have been performed on a specific corpus for our task. On the other hand, they used two biomedical corpora LLL and AIMed in which proteins are manually annotated. Therefore, a possible explanation of our results being lower than those obtained in their work, could be that our performance shows a remarkable impact of automatic entity recognition on the relation extraction task. It is foreseeable that if drug names were manually labeled in our corpus, our results will improve significantly.

The DrugDDI corpus exhibits highly imbalanced distribution (only the 10% of the examples are DDIs). We have also studied the impact of the imbalanced dataset on the performance of the kernel-based approach. We have performed three experiments: *Imbalanced* in which training and testing datasets are imbalanced, *TrainingBalanced* in which only training dataset has been balanced, and *Balanced* in which both training and testing datasets have been balanced. Experiments show that balancing the training dataset does not show increased performance with respect to using imbalanced training dataset and evaluating on imbalanced data. Regarding the last experiment, although it shows better results than those obtained in the previous experiments, we are aware that this experiment is far from representative of the real situation, in which there is a high imbalance between classes.

In the near future, we plan to extend our work in several ways. First of all, we will study other solutions for imbalanced learning such as oversampling or cost-sensitive methods. We will also evaluate each of the three contexts (before, between and after) separately in order to analyze which of the three contexts is the most discriminative for DDI extraction. The roles of the interacting drugs will be labelled in the corpus, and we will study the performance of the local kernel in their detection. We believe that the good performance achieved using shallow linguistic information provides a higher baseline, being possible to measure improvements obtained using other methods such as full syntactic or semantic information. Thus, we will define a semantic kernel considering semantic information from UMLS and other drug knowledge sources such as ATC codes obtained from DrugBank or drug families obtained by the DrugNer process. In addition, we are also discussing the possibility to apply parse tree or dependency graph kernels.

On the other hand, we will also deal with the extraction of relevant information on each interaction such as its mechanism, its relation to the doses of both drugs, its time course, the factors that alter an individual's susceptibility to it, its seriousness and severity, and the probability of its occurrence. This is a very complex challenge because it involves resolving temporal expressions, detecting events, recognizing other biomedical entities and their relationships, integrating drug knowledge information, among others. In addition, food-drug interactions, drug-disease interactions, ADRs, drug-protein interactions (that is, drug targets) and many other relationships in the pharmacological domain also deserve investigation.

To the best of our knowledge, this thesis has proposed the first integral solution for the automatic extraction of DDI from biomedical texts. The other major contribution of this thesis is the construction of the first corpus annotated for DDIs. We hope that this corpus can encourage other researches to explore new solutions for the extraction of DDIs. We have also developed a prototye integrating the different

developed techniques in this thesis to carry out our goal. We hope that our proposal contributes to the development of useful tools to assist healthcare professionals in the early detection of DDIs. However, we should point out that in an environment as sensitive as health, it is important to take into account that automated solutions can only facilitate routine tasks and serve as support, while the final decisions are up to the experts. To finish, we believe that this thesis can become a starting point for researching IE techniques applied to the pharmaceutical industry, for improving not only the patient safety, but also the drug discovery process by the identification of drug targets in the biomedical literature, among many other contributions.

Bibliography

S.T. Ahmed, D. Chidambaram, H. Davulcu, and C. Baral. IntEx: A Syntactic Role Driven Protein-Protein Interaction Extractor for Bio-Medical Text. *Linking Biological Literature, Ontologies and Databases: Mining Biological Semantics*, pages 54–61, 2005.

A. Airola, S. Pyysalo, J. Bjorne, T. Pahikkala, F. Ginter, and T. Salakoski. A graph kernel for protein-protein interaction extraction. In *Proceedings of BioNLP*, pages 1–9, 2008.

SF Altschul, TL Madden, AA Schaffer, J. Zhang, Z. Zhang, W. Miller, and DJ Lipman. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Research*, 25 (17):3389–3402, 1997.

S. Ananiadou and J. McNaught. *Text Mining for Biology and Biomedicine*. Artech House, England, 2006.

R.K. Ando. BioCreative II gene mention tagging system at IBM Watson. In *Proceedings of the Second BioCreative Challenge Evaluation Workshop*, pages 101–103. Citeseer, 2007.

R.K. Ando and T. Zhang. A framework for learning predictive structures from multiple tasks and unlabeled data. *The Journal of Machine Learning Research*, 6:1817–1853, 2005.

R. Angus, G. Robert, H. Mark, and G. Yikun. Mining clinical relationships from patient narratives. *BMC Bioinformatics*, 9, 2008.

E. APEAS. Estudio sobre la seguridad de los pacientes en Atención primaria de salud. *Madrid: Ministerio de Sanidad y Consumo*, 2008.

AR Aronson. Effective mapping of biomedical text to the UMLS Metathesaurus: the MetaMap program. In *Proceedings of the AMIA Symposium*, page 17, 2001. URL `http://metamap.nlm.nih.gov/`.

JK Aronson. Classifying drug interactions. *British Journal of Clinical Pharmacology*, 58(4):343–4, 2004a.

JK Aronson. Drug interactions-information, education, and the British National Formulary. *British Journal of Clinical Pharmacology*, 57(4):473–86, 2004b.

JK Aronson. Communicating information about drug interactions. *British Journal of Clinical Pharmacology*, 63:637–639, 2007.

M. Ashburner, CA Ball, JA Blake, D. Botstein, H. Butler, JM Cherry, AP Davis, K. Dolinski, SS Dwight, JT Eppig, et al. Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. *Nature Genetics*, 25(1):25–9, 2000. URL `http://www.geneontology.org/`.

A. Bairoch, R. Apweiler, C.H. Wu, W.C. Barker, B. Boeckmann, S. Ferro, E. Gasteiger, H. Huang, R. Lopez, M. Magrane, et al. The Universal Protein Resource (UniProt). *Nucleic Acids Research*, 2005.

Jason Baldridge. OpenNLP package, 2004. URL `http://opennlp.sourceforge.net/`.

V. Bashyam, G. Divita, DB Bennett, AC Browne, and RK Taira. A normalized lexical lookup approach to identifying UMLS concepts in free text. *Studies in health technology and informatics*, 129(Pt 1):545, 2007.

E. Beisswanger, V. Lee, J.J. Kim, D. Rebholz-Schuhmann, A. Splendiani, O. Dameron, S. Schulz, and U. Hahn. Gene Regulation Ontology (GRO): design principles and use cases. *Studies in health technology and informatics*, 136:9–14, 2008.

D.A. Benson, I. Karsch-Mizrachi, D.J. Lipman, J. Ostell, and D.L. Wheeler. GenBank. *Nucleic Acids Research*, 35(Database issue):D21, 2007.

J.A. Blake, J.E. Richardson, C.J. Bult, J.A. Kadin, and J.T. Eppig. MGD: the mouse genome database. *Nucleic Acids Research*, 31(1):193, 2003.

C. Blaschke and A. Valencia. The frame-based module of the SUISEKI information extraction system. *IEEE Intelligent Systems*, 17(2):14–20, 2002.

C. Blaschke, M.A. Andrade, C. Ouzounis, and A. Valencia. Automatic extraction of biological information from scientific text: protein-protein interactions. In *Proceedings of the International Conference Intelligent Systems for Molecular Biology*, volume 1999, pages 60–67, 1999.

O. Bodenreider. The unified medical language system (UMLS): integrating biomedical terminology. *Nucleic Acids Research*, 32(Database Issue):D267, 2004. URL http://www.nlm.nih.gov/research/umls/.

B. Boeckmann, A. Bairoch, R. Apweiler, M.C. Blatter, A. Estreicher, E. Gasteiger, M.J. Martin, K. Michoud, C. O'Donovan, I. Phan, et al. The SWISS-PROT protein knowledgebase and its supplement TrEMBL in 2003. *Nucleic Acids Research*, 31(1):365, 2003. URL http://www.expasy.ch/sprot/.

M.B. Bottorff. Statin safety and drug interactions: clinical implications. *The American Journal of Cardiology*, 97(8S1):27–31, 2006.

E. Brill. A simple rule-based part of speech tagger. In *Proceedings of the workshop on Speech and Natural Language, ACL.*, page 116, 1992.

T. Briscoe and J. Carroll. Robust accurate statistical annotation of general text. In *Proceedings of the 3rd International Conference on Language Resources and Evaluation*, pages 1499–1504, 2002.

T. Briscoe, J. Carroll, and R. Watson. The second release of the RASP system. In *Proceedings of the COLING/ACL*, volume 6, 2006.

R. Bunescu. Associative anaphora resolution: A web-based approach. In *Proceedings of the EACL-2003 Workshop on the Computational Treatment of Anaphora*, pages 47–52, 2003.

R. Bunescu and R. Mooney. Subsequence kernels for relation extraction. *Advances in Neural Information Processing Systems*, 18:171, 2006.

R. Bunescu, R. Ge, R.J. Kate, E.M. Marcotte, R.J. Mooney, A.K. Ramani, and Y.W. Wong. Comparative experiments on learning information extractors for proteins and their interactions. *Artificial Intelligence in Medicine*, 33(2):139–155, 2005.

R.C. Bunescu and R.J. Mooney. A shortest path dependency kernel for relation extraction. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 724–731, 2005.

N. Burton-Roberts. Nominal apposition. *Foundations of language*, 13(3):391–419, 1975.

M.E. Califf and R.J. Mooney. Relational learning of pattern-match rules for information extraction. In *Proceedings of the national conference on Artificial intelligence*, pages 328–334, 1999.

N. Calzolari, A. Lenci, and A. Zampolli. The EAGLES/ISLE computational lexicon working group for multilingual computational lexicons. In *Proceedings of the First International Workshop on Multimedia Annotation. Tokyo (Japan)*, 2001.

C. Cano, T. Monaghan, A. Blanco, DP Wall, and L. Peshkin. Collaborative text-annotation resource for disease-centered relation extraction from biomedical text. *Journal of Biomedical Informatics*, 2009.

J.G. Caporaso, W.A. Baumgartner, D.A. Randolph, K.B. Cohen, and L. Hunter. MutationFinder: a high-performance system for extracting point mutation mentions from text. *Bioinformatics*, 23(14):1862, 2007.

X. Carreras and L. Marquez. Filtering ranking perceptron learning for partial parsing. *Machine Learning*, 60(1):41–71, 2005.

J. Castano, J. Zhang, and J. Pustejovsky. Anaphora resolution in biomedical literature. In *International Symposium on Reference Resolution*, 2002.

C.C. Chang and C.J. Lin. LIBSVM: a library for support vector machines, 2001.

WW Chapman, W. Bridewell, P. Hanbury, GF Cooper, and BG Buchanan. Evaluation of negation phrases in narrative clinical reports. In *Proceedings of the AMIA Symposium*, page 105. American Medical Informatics Association, 2001.

E. Charniak. A maximum-entropy-inspired parser. In *Proceedings of the first conference on North American chapter of the Association for Computational Linguistics*, pages 132–139, 2000.

Y. Chen, F. Liu, and B. Manderick. Normalizing Interactor Proteins and Extracting Interaction Protein Pairs using Support Vector Machines. In *Proceedings of the BioCreative II. 5 Workshop 2009 on Digital Annotations*, page 29, 2009.

JM Cherry, C. Adler, C. Ball, SA Chervitz, SS Dwight, ET Hester, Y. Jia, G. Juvik, T. Roe, M. Schroeder, et al. SGD: Saccharomyces genome database. *Nucleic acids research*, 26(1):73, 1998.

N. Chomsky. *Syntactic structures*. Walter de Gruyter, 2002.

H.W. Chun, Y. Tsuruoka, J.D. Kim, R. Shiba, N. Nagata, T. Hishiki, and J. Tsujii. Extraction of gene-disease relations from Medline using domain dictionaries and machine learning. In *Pacific Symposium on Biocomputing*, volume 11, pages 4–15, 2006.

K.B. Cohen, L. Fox, P.V. Ogren, and L. Hunter. Corpus design for biomedical natural language processing. In *Proceedings of the ACL-ISMB workshop on linking biological literature, ontologies and databases*, pages 38–45, 2005.

N. Collier, C. Nobata, and J. Tsujii. Extracting the names of genes and gene products with a hidden Markov model. *Proceedings of the 18th conference on Computational linguistics*, 1:201–207, 2000.

M. Collins and N. Duffy. Convolution kernels for natural language. *Advances in Neural Information Processing Systems*, 1:625–632, 2002.

M.J. Collins. A new statistical parser based on bigram lexical dependencies. In *Proceedings of the 34th annual meeting on Association for Computational Linguistics*, pages 184–191, 1996.

D.P.A. Corney, B.F. Buxton, W.B. Langdon, and D.T. Jones. BioRAT: extracting biological information from full-length papers. *Bioinformatics*, 20(17):3206–3213, 2004.

N. Cristianini and J. Shawe-Taylor. *An introduction to support Vector Machines: and other kernel-based learning methods*. Cambridge Univ Pr, 2000.

A. Culotta and J. Sorensen. Dependency tree kernels for relation extraction. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, 2004.

G.O. Curme. English grammar. *New York*, 1963.

D. Cutting, J. Kupiec, J. Pedersen, and P. Sibun. A practical part-of-speech tagger. In *Proceedings of the Third Conference on Applied Natural Language Processing*, 1992.

C. de Pablo-Sánchez and P. Martínez. Building a Graph of Names and Contextual Patterns for Named Entity Classification. In *31st European Conference on Information Retrieval*, 2009.

K. Degtyarenko, P. de Matos, M. Ennis, J. Hastings, M. Zbinden, A. McNaught, R. Alcantara, M. Darsow, M. Guedj, and M. Ashburner. ChEBI: a database and ontology for chemical entities of biological interest. *Nucleic acids research*, 36(Database issue):D344, 2008. URL `http://www.ebi.ac.uk/chebi/`.

D. Demner-Fushman, S. Ananiadou, B. Cohen, J. Pestian, J Tsujii, and B. Webber. BioNLP 2008: Current Trends in Biomedical Natural Language Processing. In *Proceedings of the Workshop BioNLP*, 2008. URL `http://compbio.uchsc.edu/BioNLP2009/index.shtml`.

MC Díaz-Galiano, MT Martín-Valdivia, and LA Ureña-López. Query expansion with a medical ontology to improve a multimodal information retrieval system. *Computers in Biology and Medicine*, 2009.

S.C. Dik. *Coordination: Its implications for the theory of general linguistics.* Elsevier Science & Technology, 1968.

N. Dimililer and E. Varoglu. Recognizing biomedical named entities using SVMs: improving recognition performance with a minimal set of features. *Lecture Notes in Computer Science*, 3886:53, 2006.

J. Ding, D. Berleant, D. Nettleton, and E. Wurtele. Mining MEDLINE: Abstracts, sentences, or phrases? In *Pacific Symposium on Biocomputing 2002: Kauai, Hawaii, 3-7 January 2002*, page 326, 2002.

S. Ding, M. Huang, and X. Zhu. Semi-supervised pattern learning for extracting relations from bioscience texts. In *Proceedings of the 5th Asia-Pacific bioinformatics conference: Hong Kong, 15-17 January 2007*, page 307, 2007.

G. Divita, T. Tse, and L. Roth. Failure analysis of MetaMap transfer (MMTx). In *Medinfo 2004: Proceedings of the 11th World Congress on Medical Informatics*, page 763, 2004.

G. Doddington, A. Mitchell, M. Przybocki, L. Ramshaw, S. Strassel, and R. Weischedel. The Automatic Content Extraction (ACE) Program–Tasks, Data, and Evaluation. In *Proceedings of LREC*, pages 837–840, 2004.

R.A. Drysdale and M.A. Crosby. FlyBase: genes and gene models. *Nucleic Acids Research*, 33(Database issue):D390–D395, 2005.

S. Duda, C. Aliferis, R. Miller, A. Statnikov, and K. Johnson. Extracting drug-drug interaction articles from MEDLINE to improve the content of drug databases. In *AMIA Annual Symposium Proceedings*, volume 2005, page 216, 2005.

J.A.D.C.M. Edward, P.A.A.J.G. Philip, and D.H.R.C. Van Bergen. Concordance of Severity Ratings Provided in Four Drug Interaction Compendia. *Journal of the American Pharmaceutical Association*, 44(2), 2004.

K. Eilbeck, S. Lewis, C. Mungall, M. Yandell, L. Stein, R. Durbin, and M. Ashburner. The Sequence Ontology: a tool for the unification of genome annotations. *Genome biology*, 6(5):R44, 2005.

M.E.S. Elwood, P. St, A. Werrell, M. St, A. Board, and M. Chicago. Medical Subject Headings. *Journal Of the American Medical Association (JAMA)*, 1948.

J.H. Eom, S. Kim, S.H. Kim, B.T. Zhang, et al. A tree kernel-based method for protein-protein interaction mining from biomedical literature. *Lecture Notes in Computer Science*, 3886:42, 2006.

C. Fabricius-Hansen and W. Ramm. *'Subordination' Versus 'coordination' in Sentence and Text: A Cross-linguistic Perspective.* J. Benjamins Pub. Co., 1984.

C. Fellbaum et al. *WordNet: An electronic lexical database.* MIT press Cambridge, MA, 1998.

RE. Ferner and JK. Aronson. Communicating drug safety. *JBM*, 333:1435, 2006.

W.N. Francis. The Structure of American English. *New York*, pages 409–17, 1958.

K. Franzen, G. Eriksson, F. Olsson, L. Asker, P. Lidén, and J. Cöster. Protein names and how to find them. *International Journal of Medical Informatics*, 67(1-3):49–61, 2002.

K. Franzén, G. Eriksson, F. Olsson, L. Asker, P. Lidén, and J. Cöster. Protein names and how to find them. *International Journal of Medical Informatics*, 67(1-3):49–61, 2002.

C.C. Fries. *The structure of English: An introduction to the construction of English sentences.* Harcourt, Brace, 1952.

K. Fukuda, A. Tamura, T. Tsunoda, and T. Takagi. Toward information extraction: identifying protein names from biological papers. In *Pacific Symposium on Biocomputing*, volume 707, page 18, 1998.

K. Fundel, R. Kuffner, and R. Zimmer. RelEx–relation extraction using dependency parse trees. *Bioinformatics*, 23(3):365, 2007a.

K. Fundel, R. Kuffner, and R. Zimmer. RelEx–Relation extraction using dependency parse trees. *Bioinformatics*, 23(3):365, 2007b.

R. Gaizauskas, G. Demetriou, PJ Artymiuk, and P. Willett. Protein structures and information extraction from biological texts: the PASTA system. *Bioinformatics*, 19(1):135, 2003.

Y. Garten and R. Altman. Pharmspresso: a text mining tool for extraction of pharmacogenomic concepts and relationships from full text. *BMC bioinformatics*, 10(Suppl 2):S6, 2009.

C. Gasperin and W.G. Building. Semi-supervised anaphora resolution in biomedical texts. In *Proceedings of the BioNLP Workshop on Linking Natural Language Processing and Biology at HLT-NAACL*, volume 6, pages 96–103, 2006.

S. Gaudan, H. Kirsch, and D. Rebholz-Schuhmann. Resolving abbreviations to their senses in Medline. *Bioinformatics*, 21(18):3658–3664, 2005.

A.G. Gilman et al. *Goodman and Gilman's the pharmacological basis of therapeutics.* McGraw-Hill New York, 1992.

C. Giuliano, A. Lavelli, and L. Romano. Exploiting shallow linguistic information for relation extraction from biomedical literature. In *Proceedings of the Eleventh Conference of the European Chapter of the Association for Computational Linguistics (EACL-2006)*, pages 5–7, 2006.

C. Giuliano, A. Lavelli, D. Pighin, and L. Romano. FBK-IRST: Kernel methods for semantic relation extraction. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval 2007)*, pages 141–144, 2007a.

C. Giuliano, A. Lavelli, and L. Romano. Relation extraction and the influence of automatic named-entity recognition. *ACM Transactions on Speech and Language Processing (TSLP)*, 5(1), 2007b.

D. Grinberg, J. Lafferty, and D. Sleator. A robust parsing algorithm for link grammars. *Arxiv preprint cmp-lg/9508003*, 1995.

B.J. Grosz, S. Weinstein, and A.K. Joshi. Centering: A framework for modeling the local coherence of discourse. *Computational linguistics*, 21(2):203–225, 1995.

S. Guo and N. Ramakrishnan. Mining Linguistic Cues for Query Expansion: Applications to Drug Interaction Search. In *In CIKM '09: Proceeding of the 18th ACM conference on Information and knowledge management*, pages 335–344, 2009.

Z. GuoDong, S. Jian, Z. Jie, and Z. Min. Exploring various knowledge in relation extraction. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics (ACL)*, page 434, 2005.

H. Gurulingappa, C. Kolarik, M. Hofmann-Apitius, and J. Fluck. Concept-Based Semi-Automatic Classification of Drugs. *Journal of chemical information and modeling*, 2009.

J. Hakenberg1a, R. Leaman, S. Ha, N. Jonnalagadda, R. Sullivan, C. Miller, L. Tari, C. Baral, and G Gonzalez. Online protein interaction extraction and normalization at Arizona State University. In *Proceedings of the BioCreative II. 5 Workshop 2009 on Digital Annotations*, page 33, 2009.

D. Hanisch, J. Fluck, H.T. Mevissen, and R. Zimmer. Playing Biology's Name Game: Identifying protein names in scienfific text. *Biocomputing 2003*, 2002.

D. Hanisch, K. Fundel, H.T. Mevissen, R. Zimmer, and J. Fluck. ProMiner: rule-based protein and gene entity recognition. *BMC bioinformatics*, 6(Suppl 1):S14, 2005.

PD Hansten. Drug interaction management. *Pharmacy World & Science*, 25(3):94–97, 2003.

Y. Hao, X. Zhu, M. Huang, and M. Li. Discovering patterns to extract protein-protein interactions from the literature: Part II. *Bioinformatics*, 21(15):3294, 2005.

H. He and E.A. Garcia. Learning from Imbalanced Data. *IEEE Transactions on Knowledge and Data Engineering*, 21(9):1263, 2009.

W. Hersh and R.T. Bhupatiraju. TREC genomics track overview. In *TREC 2003*, pages 14–23, 2003. URL http://ir.ohsu.edu/genomics/.

W. Hersh, R.T. Bhuptiraju, L. Ross, P. Johnson, A.M. Cohen, and D.F. Kraemer. TREC 2004 genomics track overview. In *Proceedings of the 13th Text REtrieval Conference*, 2004.

W. Hersh, A. Cohen, J. Yang, R.T. Bhupatiraju, P. Roberts, and M. Hearst. TREC 2005 genomics track overview. In *Proceedings of the fourteenth text retrieval conference (TREC 2005)*, 2005. URL `http://ir.ohsu.edu/genomics/`.

K.M. Hettne, R.H. Stierum, M.J. Schuemie, P.J.M. Hendriksen, B.J.A. Schijvenaars, E.M. van Mulligen, J. Kleinjans, and J.A. Kors. A Dictionary to Identify Small Molecules and Drugs in Free Text. *Bioinformatics*, 2009.

L. Hirschman and N. Chinchor. MUC-7 coreference task definition. In *MUC-7 proceedings*, 1997.

L. Hirschman, A. Yeh, C. Blaschke, and A. Valencia. Overview of BioCreAtIvE: critical assessment of information extraction for biology. *BMC bioinformatics*, 6(Suppl 1):S1, 2005. URL `http://www.biocreative.org/`.

H.S. Huang, Y.S. Lin, K.T. Lin, C.J. Kuo, Y.M. Chang, B.H. Yang, I.F. Chung, and C.N. Hsu. High-recall gene mention recognition by unification of multiple backward parsing models. In *Proceedings of the Second BioCreative Challenge Evaluation Workshop*, pages 109–111. Citeseer, 2007.

M. Huang, X. Zhu, Y. Hao, D.G. Payan, K. Qu, and M. Li. Discovering patterns to extract protein-protein interactions from full texts. *Bioinformatics*, 20(18):3604–3612, 2004a.

M. Huang, X. Zhu, and M. Li. A hybrid method for relation extraction from biomedical literature. *International Journal of Medical Informatics*, 75(6):443–455, 2006a.

M. Huang, S. Ding, H. Wang, and X. Zhu. Mining physical protein-protein interactions from the literature. *Genome Biology*, 9(Suppl 2):S12, 2008.

ML Huang, XY Zhu, SL Ding, H. Yu, and M. Li. ONBIRES: Ontology-based biological relation extraction system. In *Proceedings of the Fourth Asia Pacific Bioinformatics Conference*, pages 327–336, 2006b.

W. Huang, Y. Nakamori, S. Wang, and T. Ma. Mining Medline for New Possible Relations of Concepts. *Lecture notes in computer science*, pages 794–799, 2004b.

B.L. Humphreys, D.A.B. Lindberg, H.M. Schoolman, and G.O. Barnett. The Unified Medical Language System: An Informatics Research Collaboration. *Journal of the American Medical Informatics Association*, 5(1):1–11, 1998. URL `http://www.nlm.nih.gov/research/umls/`.

K. Humphreys, G. Demetriou, and R. Gaizauskas. Two applications of information extraction to biological science journal articles: Enzyme interactions and protein structures. In *Pacific Symposium on Biocomputing*, volume 2000, pages 502–513, 2000.

CA Jankel, JA. McMillan, and BC. Martin. Effects of drug interactions on outcomes of patients receiving warfarin or theophylline. *Am J Hosp Pharm*, 51:661–666, 1994.

O. Jespersen and J.D. McCawley. *Analytic syntax*. University of Chicago Press, 1984.

J. Jiang and C.X. Zhai. A systematic exploration of the feature space for relation extraction. In *Proceedings of NAACL HLT*, pages 113–120, 2007.

J.B. Johannessen. *Coordination*. Oxford University Press, USA, 1998.

N. Kambhatla. Combining lexical, syntactic, and semantic features with maximum entropy models for extracting relations. In *Proceedings of the ACL 2004 on Interactive poster and demonstration sessions*, 2004.

M. Kanehisa and S. Goto. KEGG: Kyoto encyclopedia of genes and genomes. *Nucleic acids research*, 28 (1):27, 2000. URL `http://www.genome.jp/kegg/`.

S. Katrenko and P. Adriaans. Learning relations from biomedical corpora using dependency trees. *Lecture Notes in Computer Science*, 4366:61–80, 2007.

J. Kim, Z. Zhang, J.C. Park, and S.K. Ng. BioContrasts: extracting and exploiting protein-protein contrastive relations from biomedical literature. *Bioinformatics*, 22(5):597, 2006.

J.D. Kim, T. Ohta, Y. Tateisi, and J. Tsujii. GENIA corpus-a semantically annotated corpus for bio-textmining. *Bioinformatics*, 19(90001):180–182, 2003.

J.D. Kim, T. Ohta, Y. Tsuruoka, Y. Tateisi, and N. Collier. Introduction to the bio-entity recognition task at JNLPBA. In *Proceedings of the International Joint Workshop on Natural Language Processing in Biomedicine and its Applications*, pages 70–75, 2004.

J.D. Kim, T. Ohta, S. Pyysalo, Y. Kano, and J. Tsujii. Overview of BioNLP'09 shared task on event extraction. In *Proceedings of the Workshop on BioNLP: Shared Task*, pages 1–9, 2009.

J.J. Kim and J.C. Park. Bioar: Anaphora resolution for relating protein names to proteome database entries. In *ACL*, pages 79–86, 2004a.

JJ Kim and JC Park. Bioie: retargetable information extraction and ontological annotation of biological interactions from the literature. *Journal of bioinformatics and computational biology*, 2(3):551, 2004b.

S. Kim, J. Yoon, and J. Yang. Kernel approaches for genic interaction extraction. *Bioinformatics*, 24(1): 118, 2008.

R. Klinger, C.M. Friedrich, J. Fluck, and M. Hofmann-Apitius. Named entity recognition with combinations of conditional random fields. In *Proceedings of the Second BioCreative Challenge Evaluation Workshop*, pages 105–107. Citeseer, 2007.

LT Kohn, JM Corrigan, and MS Donaldson. To err is human: building a safer health system. Washington, DC: Institute of Medicine. *National Academy of Sciences*, 1999.

A. Koike and T. Takagi. Gene/protein/family name recognition in biomedical literature. *Proceedings of BioLINK 2004: Linking Biological Literature, Ontologies, and Databases*, pages 9–16, 2004.

C. Kolarik, M. Hofmann-Apitius, M. Zimmermann, and J. Fluck. Identification of new drug classification terms in textual resources. *Bioinformatics*, 23(13):i264, 2007.

Z. Kou, W.W. Cohen, and R.F. Murphy. High-recall protein entity recognition using a dictionary. *Bioinformatics-Oxford*, 21(1):266, 2005.

M. Krallinger and A. Valencia. Evaluating the detection and ranking of protein interaction relevant articles: the BioCreative challenge interaction article sub-task (IAS). In *Proceedings of the Second Biocreative Challenge Evaluation Workshop*, 2007.

M. Krallinger, F. Leitner, C. Rodriguez-Penagos, and A. Valencia. Overview of the protein-protein interaction annotation extraction task of BioCreative II. *Genome Biology*, 9(Suppl 2):S4, 2008. URL `http://www2.informatik.hu-berlin.de/ hakenber/corpora/`.

M. Krallinger, F. Leitner, and A. Valencia. The BioCreative II.5 challenge overview. In *Proceedings of the BioCreative II. 5 Workshop 2009 on Digital Annotations*, page 19, 2009.

M. Krauthammer, A. Rzhetsky, P. Morozov, and C. Friedman. Using BLAST for identifying gene and protein names in journal articles. *Gene*, 259(1-2):245–252, 2000.

C.J. Kuo, Y.M. Chang, H.S. Huang, K.T. Lin, B.H. Yang, Y.S. Lin, C.N. Hsu, and I.F. Chung. Rich feature set, unification of bidirectional parsing and dictionary filtering for high F-score gene mention tagging. In *Proc of the Second BioCreative Challenge Evaluation Workshop (BioCreative II) Madrid, Spain*. Citeseer, 2007.

M.V. LAM, G.M. MCCART, and C. TSOUROUNIS. An assessment of Free, online drug-drug interaction screening programs(DSPs). *Hospital pharmacy(Philadelphia, PA)*, 38(7):662–668, 2003.

A. Lavelli, M.E. Califf, F. Ciravegna, D. Freitag, C. Giuliano, N. Kushmerick, and L. Romano. A critical survey of the methodology for IE evaluation. In *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC 2004)*, pages 1655–1658, 2004.

R. Leaman and G. Gonzalez. Banner: An executable survey of advances in biomedical named entity recognition. In *Pacific Symposium on Biocomputing*, volume 13, pages 652–663, 2008.

L. Leape and D. Berwick. Five years alter to err is human. what have we learned. *Journal Of the American Medical Association (JAMA)*, 293:2384–2390, 2005.

M. Lease and E. Charniak. Parsing Biomedical Literature. *corpora*, 6(7):8–9, 2005.

K.J. Lee, Y.S. Hwang, and H.C. Rim. Two-phase biomedical NE recognition based on SVMs. In *Proceedings of the ACL 2003 workshop on Natural language processing in biomedicine*, volume 13, pages 33–40, 2003.

G. Leech. Corpus annotation schemes. *Literary and linguistic computing*, 8(4):275–281, 1993.

G. Leroy, H. Chen, and J.D. Martinez. A shallow parser based on closed-class words to capture relations in biomedical text. *Journal of Biomedical Informatics*, 36(3):145–158, 2003.

J. Li, Z. Zhang, X. Li, and H. Chen. Kernel-based learning for biomedical relation extraction. *Journal of the American Society for Information Science and Technology*, 59(5), 2008.

Q. Li and Y.F.B. Wu. Identifying important concepts from medical documents. *Journal of Biomedical Informatics*, 39(6):668–679, 2006.

T. Liang and Y. Lin. Anaphora Resolution for Biomedical Literature by Exploiting Multiple Resources. *Lecture notes in computer science*, 3651:742, 2005.

D. Lin. MINIPAR: a minimalist parser. In *Maryland Linguistics Colloquium*, 1999.

Y. Lin, T. Liang, and T. Hsinchu. Pronominal and sortal anaphora resolution for biomedical literature. In *Proceedings of ROCLING XVI: Conference on Computational Linguistics and Speech Processing*, 2004.

C.E. Lipscomb. Medical subject headings (MeSH). *Bulletin of the Medical Library Association*, 88(3): 265, 2000. URL http://www.ncbi.nlm.nih.gov/mesh.

Y. Liu, Z. Shi, and A. Sarkar. Exploiting rich syntactic information for relation extraction from biomedical articles. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Companion Volume, Short Papers on XX*, pages 97–100, 2007.

H. Lodhi, C. Saunders, J. Shawe-Taylor, N. Cristianini, and C. Watkins. Text classification using string kernels. *The Journal of Machine Learning Research*, 2:419–444, 2002.

D. Longo, J. Hewett, B. Ge, and S. Schubert. The long road to patient safety. *Journal Of the American Medical Association (JAMA)*, 294(22):2858–2865, 2005.

A. Lourenço, R. Carreira, S. Carneiro, P. Maia, D. Glez-Peña, F. Fdez-Riverola, E.C. Ferreira, I. Rocha, and M. Rocha. @ Note: a workbench for biomedical text mining. *Journal of Biomedical Informatics*, 42(4):710–720, 2009.

D. Maglott, J.S. Amberger, and A. Hamosh. Online Mendelian Inheritance in Man (OMIM): A Directory of Human Genes and Genetic Disorders, 2002. URL http://www.ncbi.nlm.nih.gov/sites/entrez?db=omim.

D. Maglott, J. Ostell, K.D. Pruitt, and T. Tatusova. Entrez Gene: gene-centered information at NCBI. *Nucleic acids research*, 2006. URL http://www.ncbi.nlm.nih.gov/gene/.

A.T. McCray, S. Srinivasan, and A.C. Browne. Lexical methods for managing variation in biomedical terminologies. In *Annual Symposium on Computer Application in Medical Care*, volume 18, pages 235–235, 1994. URL http://ii.nlm.nih.gov/MTI/phrasex.shtml.

F. Meng, L.W. D´volio, A.A. Chen, R.K. Taira, and H. Kangarloo. Generating Models of Surgical Procedures using UMLS Concepts and Multiple Sequence Alignment. In *AMIA Annual Symposium Proceedings*, volume 2005, page 520, 2005.

S. Meystre and P.J. Haug. Natural language processing to extract medical problems from electronic clinical documents: performance evaluation. *Journal of biomedical informatics*, 39(6):589–599, 2006.

S. Mika and B. Rost. NLProt: extracting protein names and sequences from papers. *Nucleic acids research*, 32(Web Server Issue):W634, 2004.

S. Miyazaki, H. Sugawara, K. Ikeo, T. Gojobori, and Y. Tateno. DDBJ in the stream of various biological data. *Nucleic acids research*, 32(Database Issue):D31, 2004. URL http://www.ddbj.nig.ac.jp/searches-e.html.

R. Morante and W. Daelemans. Learning the scope of hedge cues in biomedical texts. In *Proceedings of the Workshop on BioNLP*, pages 28–36, 2009.

H.M. Muller, E.E. Kenny, and P.W. Sternberg. Textpresso: an ontology-based information retrieval and extraction system for biological literature. *PLoS Biol*, 2(11):e309, 2004a.

H.M. Muller, E.E. Kenny, and P.W. Sternberg. Textpresso: an ontology-based information retrieval and extraction system for biological literature. *PLoS Biol*, 2(11):e309, 2004b.

P.G. Mutalik, A. Deshpande, and P.M. Nadkarni. 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, 2001.

M. Naguib, MM. Magboul, and R. Jaroudi. Clinically significant drug interactions with general anesthetics–incidence, mechanisms and management. *Middle East J Anesthesiol*, 14:127–83, 1997.

NCBI. NCBI National Center for Biotechnology Information Taxonomy. URL http://www.ncbi.nlm.nih.gov/sites/entrez?db=taxonomy.

C. Nedellec. Learning language in logic-genic interaction extraction challenge. In *Proceedings of the ICML05 workshop: Learning Language in Logic (LLL05)*, volume 18, pages 97–99, 2005.

G. Nenadic, I. Spasic, and S. Ananiadou. Terminology-driven mining of biomedical literature. *Bioinformatics*, 19(8):938.

V. Ng. Supervised ranking for pronoun resolution: Some recent improvements. In *Proceedings of the National Conference on Artificial Intelligence*, volume 20, page 1081, 2005.

N.L.T. Nguyen and J.D. Kim. Exploring Domain Differences for the Design of Pronoun Resolution Systems for Biomedical Text. In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, pages 625–632, 2008.

V.V. Nguyen, M.L. Nguyen, and A. Shimazu. Clause Splitting with Conditional Random Fields. *Information and Media Technologies*, 4(1):57–75, 2009.

AS Nies. *Principles of the therapeutics. The Pharmacological Basis of Therapeutics*, pages 45–66. 2001.

S. Novichkova, S. Egorov, and N. Daraselia. MedScan, a natural language processing engine for MEDLINE abstracts. *Bioinformatics*, 19(13):1699–1706, 2003.

P.V. Ogren. Knowtator: a plug-in for creating training and evaluation data sets for biomedical natural language systems. In *Presentation Abstracts*, page 73, 2006.

T. Ohta, Y. Tateisi, and J.D. Kim. The GENIA corpus: An annotated research abstract corpus in molecular biology domain. In *Proceedings of the second international conference on Human Language Technology Research*, pages 82–86, 2002.

T. Ono, H. Hishigaki, A. Tanigami, and T. Takagi. Automated extraction of information on protein-protein interactions from the biological literature, 2001.

E. Pafilis, S.I. O'Donoghue, L.J. Jensen, H. Horn, M. Kuhn, N.P. Brown, and R. Schneider. Reflect: augmented browsing for the life scientist. *Nature Biotechnology*, 27(6):508–510, 2009.

T.M. Phuong, D. Lee, and K.H. Lee. Learning Rules to Extract Protein Interactions from Biomedical Text. *Advances in Knowledge Discovery and Data Mining: 7th Pacific-Asia Conference, Pakdd 2003, Seoul. Korea, April 30-May 2, 2003: Proceedings*, 2003.

M. Pirmohamed, S. James, S. Meakin, C. Green, A.K. Scott, T.J. Walley, K. Farrar, B.K. Park, and A.M. Breckenridge. Adverse drug reactions as cause of admission to hospital: prospective analysis of 18 820 patients. *British Medical Journal*, 329(7456):15–19, 2004.

BOT Plus. Base de datos del conocimiento sanitario, 2008.

M. Poesio and M.A. Kabadjov. A general-purpose, off-the-shelf anaphora resolution module: Implementation and preliminary evaluation. In *Proceedings of the 4th International Conference on Language Resources and Evaluation*, 2004.

M. Poesio, T. Ishikawa, S.S. im Walde, and R. Viera. Acquiring lexical knowledge for anaphora resolution. In *Proceedings of the 3rd Conference on Language Resources and Evaluation (LREC)*, 2002.

N. Ponomareva, F. Pla, A. Molina, and P. Rosso. Biomedical named entity recognition: A poor knowledge hmm-based approach. *Lecture Notes in Computer Science*, 4592:382, 2007.

M. Poprat and U. Hahn. Quantitative Data on Referring Expressions in Biomedical Abstracts. In *Proceedings of the Workshop on BioNLP 2007, (ACL 2007)*, 2007.

M.F. Porter. An algorithm for su x stripping. *Program*, 14(3):130–137, 1980.

S. Povey, R. Lovering, E. Bruford, M. Wright, M. Lush, and H. Wain. The HUGO gene nomenclature committee (HGNC). *Human genetics*, 109(6):678–680, 2001. URL http://www.genenames.org/.

K.D. Pruitt and D.R. Maglott. RefSeq and LocusLink: NCBI gene-centered resources. *Nucleic Acids Research*, 29(1):137, 2001.

J. Pustejovsky, J. Castano, R. Saurí, A. Rumshinsky, J. Zhang, and W. Luo. Medstract: creating large-scale information servers for biomedical libraries. In *Proceedings of the ACL-02 workshop on Natural language processing in the biomedical domain-Volume 3*, pages 85–92, 2002a.

J. Pustejovsky, J. Castano, R. Saurí, A. Rumshinsky, J. Zhang, and W. Luo. Medstract: creating large-scale information servers for biomedical libraries. In *Proceedings of the ACL-02 workshop on Natural language processing in the biomedical domain-Volume 3*, page 92, 2002b.

J. Pustejovsky, J. Castano, J. Zhang, M. Kotecki, and B. Cochran. Robust relational parsing over biomedical literature: extracting inhibit relations. In *Pacific Symposium on Biocomputing*, volume 362, page 73, 2002c.

S. Pyysalo, F. Ginter, J. Heimonen, J. Björne, J. Boberg, J. Järvinen, and T. Salakoski. BioInfer: a corpus for information extraction in the biomedical domain. *BMC bioinformatics*, 8(1):50, 2007.

R. Quirk, S. Greenbaum, and G. Leech. *Comprehensive Grammar of the English Language.* London: Longman, 1985.

A. Ratnaparkhi. Mxpost (maximum entropy pos tagger), ver. 1.0, 1996. URL http://www.cis.upenn.edu/ dbikel/software.html.

D. Rebholz-Schuhmann, A. Jimeno-Yepes, E. van Mulligen, N. Kang, J. Kors, D. Milward, P. Corbett, and U. Hahn. The CALBC Silver Standard Corpus - Harmonizing multiple semantic annotations in a large biomedical corpus. In *The 3rd International Symposium on Languages in Biology and Medicine*, 2009.

L.H. Reeve, H. Han, and A.D. Brooks. The use of domain-specific concepts in biomedical text summarization. *Information Processing and Management*, 43(6):1765–1776, 2007.

A. Refoufi. A Multiple Knowledge Sources Algorithm for Anaphora Resolution. *Asian Journal of Information Technology*, 5(1):48–53, 2006.

F. Reichartz, I. Fraunhofer, G. St Augustin, H. Korte, and G. Paass. Composite Kernels For Relation Extraction. In *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*, pages 365–368, 2009.

J. Renders. Kernel Methods in Natural Language Processing, 2004.

F. Rinaldi, G. Schneider, K. Kaljurand, J. Dowdall, C. Andronis, A. Persidis, and O. Konstanti. Mining relations in the GENIA corpus. *Proceedings of the Second European Workshop on Data Mining and Text Mining for Bioinformatics*, pages 61–68, 2004.

F. Rinaldi, G. Schneider, K. Kaljurand, M. Hess, C. Andronis, O. Konstandi, and A. Persidis. Mining of relations between proteins over biomedical scientific literature using a deep-linguistic approach. *Artificial Intelligence In Medicine*, 39(2):127–136, 2007.

A. Roberts, R. Gaizauskas, M. Hepple, N. Davis, G. Demetriou, Y. Guo, J.S. Kola, I. Roberts, A. Setzer, A. Tapuria, et al. The CLEF corpus: semantic annotation of clinical text. In *AMIA Annual Symposium Proceedings*, volume 2007, page 625, 2007.

A. Roberts, R. Gaizauskas, M. Hepple, G. Demetriou, Y. Guo, I. Roberts, and A. Setzer. Building a semantically annotated corpus of clinical texts. *Journal of Biomedical Informatics*, 42(5):950–966, 2009.

A. Rodríguez-Terol, C. Camacho, et al. Calidad estructural de las bases de datos de interacciones. *Farmacia Hospitalaria*, 33(03):134, 2009.

B. Rosario and M. Hearst. Classifying semantic relations in bioscience texts. *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics, ACL. Barcelona, Spain*, 2004a.

B. Rosario and M.A. Hearst. Classifying semantic relations in bioscience texts. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, 2004b.

JU Rosholm, L. Bjerrum, J. Hallas, J. Worm, and LF Gram. Polypharmacy and the risk of drug-drug interactions among Danish elderly. A prescription database study. *Danish medical bulletin*, 45(2):210, 1998.

D.L. Rubin, C.F. Thorn, T.E. Klein, and R.B. Altman. A statistical approach to scanning the biomedical literature for pharmacogenetics knowledge. *Journal of the American Medical Informatics Association*, 12(2):121–129, 2005.

R.G.G. Russell. Determinants of structure–function relationships among bisphosphonates. *Bone*, 40 (5S2):21–25, 2007.

R. Sætre, K. Yoshida, M. Miwa, T Matsuzaki, Y. Kano, and J. Tsujii. AkaneRE Relation Extraction: Protein Normalization (INT) and Interaction (IPT) in the BioCreative II.5 Challenge. In *Proceedings of the BioCreative II. 5 Workshop 2009 on Digital Annotations*, page 33, 2009.

K. Sagae. Dependency parsing and domain adaptation with LR models and parser ensembles. In *Proceedings of the Eleventh Conference on Computational Natural Language Learning*, 2007. URL http://www.cs.cmu.edu/ sagae/parser/gdep/.

O. Sánchez, M. Poesio, M.A. Kabadjov, and R. Tesar. What kind of problems do protein interactions raise for anaphora resolution? A preliminary analysis. *Proceedings of the second SMBM 2006*, pages 109–112, 2006.

O. Sánchez-Graillet and M. Poesio. Negation of protein protein interactions: analysis and extraction. *Bioinformatics*, 23(13):i424, 2007.

S. Sarawagi. Information extraction. *Foundations and Trends® in Databases*, 1(3):261–377, 2007.

G. Schneider, K. Kaljurand, F. Rinaldi, and T. Kuhn. Pro3Gres parser in the CoNLL domain adaptation shared task. In *Proceedings of the CoNLL Shared Task Session of EMNLP-CoNLL*, pages 1161–1165, 2007.

M.J. Schuemie, B. Mons, M. Weeber, and J.A. Kors. Evaluation of techniques for increasing recall in a dictionary approach to gene and protein name identification. *Journal of Biomedical Informatics*, 40 (3):316 – 324, 2007.

C. Seiffert, T.M. Khoshgoftaar, and J. Van Hulse. Hybrid sampling for imbalanced data. *Integrated Computer-Aided Engineering*, 16(3):193–210, 2009.

K. Seki and J. Mostafa. A hybrid approach to protein name identification in biomedical texts. *Information Processing and Management*, 41(4):723–743, 2005.

T. Sekimizu, HS Park, and J. Tsujii. Identifying the Interaction between Genes and Gene Products Based on Frequently Seen Verbs in Medline Abstracts. In *Genome informatics. Workshop on Genome Informatics*, volume 9, page 62, 1998.

J. Shawe-Taylor and N. Cristianini. *Kernel methods for pattern analysis*. Cambridge Univ Pr, 2004.

A. Siddharthan. Syntactic simplification and text cohesion. *Research on Language & Computation*, 4(1): 77–109, 2006.

E. Sirohi and P. Peissig. Study of Effect of Drug Lexicons on Medication Extraction from Electronic Medical Records. *Biocomputing 2005: Proceedings of the Pacific Symposium, Hawaii, USA 4-8 January 2005*, 2004.

D.D.K. Sleator and D. Temperley. Parsing English with a link grammar. *Arxiv preprint cmp-lg/9508004*, 1995.

L. Smith, L. Tanabe, R. Ando, C.J. Kuo, I.F. Chung, C.N. Hsu, Y.S. Lin, R. Klinger, C. Friedrich, K. Ganchev, et al. Overview of BioCreative II gene mention recognition. *Genome Biology*, 9(Suppl 2): S2, 2008.

M. Song, D. Lee, and J Huan. Second International Workshop on Data and Text Mining in Bioinformatics (DTMBio) 2008. *BMC Bioinformatics*, 10:110, 2009. URL `http://biosoft.kaist.ac.kr/dtmbio2009/home.html`.

KA Spackman, KE Campbell, RA CÃ, et al. SNOMED RT: a reference terminology for health care. In *Proceedings of the AMIA Annual Fall Symposium*, page 640, 1997. URL `http://www.ihtsdo.org/snomed-ct/`.

IH Stockley. *Stockleys Drug Interaction*. Pharmaceutical Press, 2007.

G. Szarvas, V. Vincze, R. Farkas, and J. Csirik. The BioScope corpus: annotation for negation, uncertainty and their scope in biomedical texts. In *Proceedings of the Workshop on Current Trends in Biomedical Natural Language Processing, ACL*, pages 38–45, 2008.

K. Takeuchi and N. Collier. Bio-medical entity extraction using support vector machines. *Artificial Intelligence in Medicine*, 33(2):125–137, 2005.

L. Tanabe and W.J. Wilbur. Tagging gene and protein names in biomedical text, 2002.

L. Tanabe, N. Xie, L. Thom, W. Matten, and W.J. Wilbur. GENETAG: a tagged corpus for gene/protein named entity recognition. *BMC bioinformatics*, 6(Suppl 1):S3, 2005.

Y. Tateisi, Y. Miyao, K. Sagae, and J. Tsujii. GENIA-GR: a Grammatical Relation Corpus for Parser Evaluation in the Biomedical Domain. In *Proceedings of LREC*, 2008.

D.S. Tatro. *Drug interaction facts*. Facts and Comparisons, St. Louis, 2003.

J.M. Temkin and M.R. Gilder. Extraction of protein interaction information from unstructured text using a context-free grammar, 2003.

P. Thompson, S.A. Iqbal, J. McNaught, and S. Ananiadou. Construction of an annotated corpus to support biomedical information extraction. *BMC Bioinformatics*, 10(1):349, 2009.

E.F. Tjong, K. Sang, and H. Déjean. Introduction to the CoNLL-2001 shared task: clause identification. In *Proceedings of the 2001 workshop on Computational Natural Language Learning-Volume 7*, 2001.

P. Tomasulo. ChemIDplus-super source for chemical and drug information. *Medical reference services quarterly*, 21(1):53–60, 2002. URL `http://chem.sis.nlm.nih.gov/chemidplus/`.

M. Torii and H. Liu. Headwords and Suffixes in Biomedical Names. *Lecture Notes in Computer Science*, 3886:29, 2006.

M. Torii, S. Kamboj, and K. Vijay-Shanker. Using name-internal and contextual features to classify biological terms. *Journal of Biomedical Informatics*, 37(6):498–511, 2004.

R.L. Trask. *Key concepts in language and linguistics*. Routledge, 1999.

Y. Tsuruoka and J. Tsujii. Improving the performance of dictionary-based approaches in protein name recognition. *Journal of Biomedical Informatics*, 37(6):461–470, 2004a.

Y. Tsuruoka and J. Tsujii. Improving the performance of dictionary-based approaches in protein name recognition. *Journal of Biomedical Informatics*, 37(6):461–470, 2004b.

Y. Tsuruoka, Y. Tateishi, J. Kim, T. Ohta, J. McNaught, S. Ananiadou, and J. Tsujii. Developing a robust part-of-speech tagger for biomedical text. *Lecture notes in computer science*, 3746:382, 2005.

J. Van Hulse and T. Khoshgoftaar. Knowledge discovery from imbalanced and noisy data. *Data & Knowledge Engineering*, 2009.

J. Van Hulse, T.M. Khoshgoftaar, and A. Napolitano. An empirical comparison of repetitive undersampling techniques. In *Proceedings of the 10th IEEE international conference on Information Reuse & Integration*, pages 29–34, 2009.

R.R. Van Oirsouw. *The syntax of coordination*. Routledge Kegan & Paul, 1987.

K. Verspoora, C. Roeder, H. Johnson, K. Cohen, W. Baumgartner, and L. Hunter. Information Extraction of Normalized Protein Interaction Pairs Utilizing Linguistic and Semantic Cues. In *Proceedings of the BioCreative II. 5 Workshop 2009 on Digital Annotations*, page 37, 2009.

T.B. Vicedo and C.P.M. Conde. Aplicación de las nuevas tecnologías a la farmacia hospitalaria en España. *Farmacia Hospitalaria*, 31(1):17, 2007.

A. Vlachos. Evaluating and combining biomedical named entity recognition systems. In *Proceedings of the workshop on BioNLP 2007: Biological, translational, and clinical language processing. ACL 2007*, pages 199–206, 2007.

A. Vlachos and C. Gasperin. Bootstrapping and evaluating named entity recognition in the biomedical domain. In *Proceedings of the Workshop on Linking Natural Language Processing and Biology: Towards Deeper Biological Literature Analysis*, pages 138–145, 2006.

N. von Hentig. Lopinavir/Ritonavir: Appraisal of its use in HIV Therapy. *Drugs of Today*, 43(4):221–247, 2007.

A. Voutilainen and J. Heikkila. An English constraint grammar (ENGCG): a surface-syntactic parser of English. *Creating and Using English language corpora, Rodopi, Amsterdam, Atlanta*, pages 189–199, 1993.

WHO. *The Importance of Pharmacovigilance: Safety Monitoring of Medicinal Products*. World Health Organization, 2002.

WHO. Introduction to drug utilization research, 2003.

WHO. The use of stems in the selection of International Nonproprietary Names (INN) for pharmaceutical substances. *World Health Organization Press*, WHO/PSM/QSM/2006.3, 2006. URL `http://www.who.int/medicines/services/inn/en/index.html`.

J. Wilbur, L. Smith, and L. Tanabe. Biocreative 2. gene mention task. In *Proceedings of the second biocreative challenge evaluation workshop*, pages 7–16, 2007.

E. Williams. Across-the-board rule application. *Linguistic Inquiry*, pages 31–43, 1978.

J. Wingersky, J. Boerner, and D. Holguin-Balogh. *Writing paragraphs and essays: Integrating reading, writing, and grammar skills*. 2008.

D.S. Wishart, C. Knox, A.C. Guo, S. Shrivastava, M. Hassanali, P. Stothard, Z. Chang, and J. Woolsey. DrugBank: a comprehensive resource for in silico drug discovery and exploration. *Nucleic Acids Research*, 2006. URL `http://www.drugbank.ca`.

D.S. Wishart, C. Knox, A.C. Guo, D. Cheng, S. Shrivastava, D. Tzur, B. Gautam, and M. Hassanali. DrugBank: a knowledgebase for drugs, drug actions and drug targets. *Nucleic acids research*, 2007a. URL `http://www.drugbank.ca`.

D.S. Wishart, D. Tzur, C. Knox, R. Eisner, A.C. Guo, N. Young, D. Cheng, K. Jewell, D. Arndt, S. Sawhney, et al. HMDB: the human metabolome database. *Nucleic acids research*, 35(Database issue):D521, 2007b. URL `http://www.hmdb.ca/`.

D.S. Wu and T. Liang. Zero anaphora resolution by case-based reasoning and pattern conceptualization. *Expert Systems With Applications*, 36(4):7544–7551, 2009.

Y.C. Wu, T.K. Fan, Y.S. Lee, and S.J. Yen. Extracting named entities using support vector machines. *Lecture Notes in Computer Science*, 3886:91, 2006.

J. Xiao, J. Su, G.D. Zhou, and C.L. Tan. Protein-protein interaction extraction: a supervised learning approach. In *Proc Symp on Semantic Mining in Biomedicine*, pages 51–59, 2005.

R. Xu, K. Supekar, A. Morgan, A. Das, and A. Garber. Unsupervised Method for Automatic Construction of a Disease Dictionary from a Large Free Text Collection. In *AMIA Annual Symposium Proceedings*, volume 2008, page 820. American Medical Informatics Association, 2008.

Z. Yang, H. Lin, and Y. Li. Exploiting the performance of dictionary-based bio-entity name recognition in biomedical literature. *Computational Biology and Chemistry*, 2008.

Z. Yang, H. Lin, and Y. Li. BioPPISVMExtractor: A protein–protein interaction extractor for biomedical literature using SVM and rich feature sets. *Journal of Biomedical Informatics*, 2009a.

Z. Yang, H. Lin, and B. Wu. BioPPIExtractor: A protein–protein interaction extraction system for biomedical literature. *Expert Systems with Applications*, 36(2P1):2228–2233, 2009b.

A. Yeh, L. Hirschman, and A. Morgan. Background and overview for KDD Cup 2002 task 1: Information extraction from biomedical articles. *ACM SIGKDD Explorations Newsletter*, 4(2):87–89, 2002.

A. Yeh, A. Morgan, M. Colosimo, and L. Hirschman. BioCreAtIvE task 1A: gene mention finding evaluation. *BMC bioinformatics*, 6(Suppl 1):S2, 2005.

H. Yu. Towards answering biological questions with experimental evidence: automatically identifying text that summarize image content in full-text articles. *AMIA Annu Symp Proc*, pages 834–838, 2006.

D. Zelenko, C. Aone, and A. Richardella. Kernel methods for relation extraction. *The Journal of Machine Learning Research*, 3:1083–1106, 2003.

M. Zhang, J. Zhang, J. Su, and G. Zhou. A composite kernel to extract relations between entities with both flat and structured features. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 825–832, 2006.

M. Zhang, G.D. Zhou, and A. Aw. Exploring syntactic structured features over parse trees for relation extraction using kernel methods. *Information Processing and Management*, 44(2):687–701, 2008.

D. Zhou and Y. He. Extracting interactions between proteins from the literature. *Journal of Biomedical Informatics*, 2007.

G. Zhou, J. Zhang, J. Su, D. Shen, and C. Tan. Recognizing names in biomedical texts: a machine learning approach. *Bioinformatics*, 20(7):1178–1190, 2004.