UNIVERSIDADE DE LISBOA
FACULDADE DE CIÊNCIAS
DEPARTAMENTO DE INFORMÁTICA

Identifying Interactions Between Chemical Entities in Text

André Francisco Martins Lamúrias

DISSERTAÇÃO
MESTRADO EM BIOINFORMÁTICA E BIOLOGIA COMPUTACIONAL
ESPECIALIZAÇÃO EM BIOINFORMÁTICA

2014

UNIVERSIDADE DE LISBOA
FACULDADE DE CIÊNCIAS
DEPARTAMENTO DE INFORMÁTICA

Identifying Interactions Between Chemical Entities in Text

André Francisco Martins Lamúrias

DISSERTAÇÃO
MESTRADO EM BIOINFORMÁTICA E BIOLOGIA COMPUTACIONAL
ESPECIALIZAÇÃO EM BIOINFORMÁTICA

Tese orientada pelo Prof. Doutor Francisco José Moreira Couto

2014

Resumo

Novas interações entre compostos químicos são geralmente descritas em artigos científicos, os quais estão a ser publicados a uma velocidade cada vez maior. No entanto, estes artigos são dirigidos a humanos, escritos em linguagem natural, e não são processados facilmente por um computador. Métodos de prospeção de texto são uma solução para este problema, extraindo automaticamente a informação relevante da literatura. Estes métodos devem ser adaptados ao domínio e tarefa a que vão ser aplicados.

Esta dissertação propõe um sistema para identificação automática e eficaz de interações entre entidades químicas em documentos biomédicos. O sistema foi desenvolvido em dois módulos. O primeiro módulo reconhece as entidades químicas que são mencionadas num dado texto. Este módulo foi baseado num sistema já existente, o qual foi melhorado com um novo tipo de medidas de semelhança semântica. O segundo módulo identifica os pares de entidades que representam uma interação química no mesmo texto, com recurso a técnicas de Aprendizagem Automática e conhecimento específico ao domínio. Cada módulo foi avaliado separadamente, obtendo valores de precisão elevados em dois padrões de teste diferentes. Os dois módulos constituem o sistema IICE, que pode ser usado para analisar qualquer documento biomédico, de forma a encontrar entidades e interações químicas. Este sistema está acessível através de uma ferramenta *web*.

Palavras Chave: Prospeção de Texto, Aprendizagem Automática, Reconhecimento de Entidades, Extração de Relações, Semelhança Semântica

Abstract

Novel interactions between chemical compounds are often described in scientific articles, which are being published at an unprecedented rate. However, these articles are directed to humans, written in natural language, and cannot be easily processed by a machine. Text mining methods present a solution to this problem, by automatically extracting the relevant information from the literature. These methods should be adapted to the specific domain and task they are going to be applied to.

This dissertation proposes a system for automatic and efficient identification of interactions between chemical entities from biomedical documents. This system was developed in two modules. The first module recognizes the chemical entities that are mentioned in a given text. This module was based on an existing framework, which was improved with a novel type of semantic similarity measure. The second module identifies the pairs of entities that represent a chemical interaction in the same text, using Machine Learning techniques and domain knowledge. Each module was evaluated separately, achieving high precision values against two different gold standards. The two modules were constitute the IICE system, which can be used to analyze any biomedical document for chemical entities and interactions, accessible via a web tool.

Keywords: Text Mining, Machine Learning, Named Entity Recognition, Relation Extraction, Semantic Similarity

Resumo Alargado

Diariamente, é gerada uma grande quantidade de informação biomédica, disponível para a comunidade científica. Esta informação pode ter uma estrutura de dados definida, facilitando o processamento por um computador. No entanto, grande parte da informação disponibilizada está na forma de texto, sem qualquer estrutura de dados subjacente. A literatura científica é direcionada para humanos, o que torna mais difícil o processamento por um computador. Por esta razão, é necessário desenvolver métodos de prospeção que transformem o texto numa estrutura de dados. Com este tipo de métodos, é possível extrair do texto certo tipo de informações, como por exemplo, referências a interações entre entidades relevantes.

As interações químicas extraídas automaticamente de textos científicos podem ser usadas por peritos para, por exemplo, desenvolver bases de dados, ou encontrar potenciais efeitos adversos entre fármacos. Ao extrair interações de um grande conjunto de artigos, é possível que sejam encontradas interações implícitas entre compostos químicos. Se dois compostos químicos tiverem uma interação em comum, encontrada em trabalhos de investigação diferentes, com um terceiro composto, é provável que estes constituam também uma interação. O desenvolvimento de técnicas de prospeção de texto permite que este tipo de interações seja encontrado muito mais rapidamente do que uma abordagem manual.

Aprendizagem Automática consiste num conjunto de algoritmos para treinar classificadores que consigam classificar novos dados, aprendendo com um conjunto de dados anotado por peritos no domínio em que o classificador vai ser aplicado. Este tipo de abordagem tem a vantagem de se adaptar mais facilmente a novos domínios do que abordagens baseadas em dicionários ou regras fixas. Os algoritmos de

Aprendizagem Automática têm sido aplicados com sucesso em várias tarefas de prospeção de texto. Uma destas tarefas é o reconhecimento de entidades, que consiste em identificar as entidades relevantes mencionadas num dado texto. Outra tarefa, que é geralmente sequencial à anterior, consiste em extrair relação entre entidades que são descritas no texto. O objetivo é classificar se cada par de entidades é uma interação ou não, e se for, de que tipo. Estas duas tarefas têm sido aplicadas a vários domínios ao longo dos anos, sendo que o principal é geralmente textos jornalísticos.

Vários tipos de interações podem ser extraídas de documentos biomédicos, como por exemplo, proteína-proteína, doença-tratamento, e doença-gene. No domínio dos compostos químicos, algum trabalho tem sido desenvolvido para a extração de interações do tipo fármaco-fármaco. Neste sentido, foi organizada uma competição, inserida no SemEval 2013, para extração de interações deste tipo, denominada DDI Extraction. Esta foi a segunda edição desta competição, que foi dividida em duas subtarefas: a primeira consistiu na extração de entidades químicas do texto, e a segunda na identificação de interações. Seis equipas submeteram resultados para a primeira subtarefa, enquanto que oito equipas submeteram para a segunda. No entanto, apenas duas equipas submeteram resultados para as duas subtarefas. Isto mostra que é necessário mais investigação em sistemas que extraiam interações entre compostos químicos a partir de textos sem qualquer anotação prévia.

A técnicas de prospeção de texto devem ser adaptadas ao domínio ao qual vão ser aplicadas através de conjuntos de dados de treino e processos de validação dos resultados. Dois conjuntos de dados para entidades químicas foram lançados recentemente, no âmbito da tarefa CHEMDNER da competição BioCreative IV, e da tarefa DDI (*Drug-Drug Interaction*) Extraction, da competição SemEval 2013. Estes conjuntos de dados servem para treinar classificadores, e depois avaliar os resultados obtidos com o sistema desenvolvido, comparando

com outros sistemas semelhantes. Existem também bases de dados e ontologias que podem ser usadas para validar resultados obtidos com prospeção de texto. A ideia é complementar os algoritmos de Aprendizagem Automática com esta informação específica, para treino dos classificadores ou mapeamento das entidades reconhecidas a identificadores únicos. Algumas fontes de informação úteis para compostos químicos são o ChEBI (*Chemical Entities of Biological Interest*), Gene Ontology, e DrugBank.

O objetivo desta dissertação foi desenvolver um sistema para extração automática e eficaz de interações químicas de textos biomédicos. O sistema desenvolvido, chamado IICE, é baseado em algoritmos de Aprendizagem Automática, bem como recursos específicos ao domínio biomédico. O sistema IICE é constituído por dois módulos, que foram desenvolvidos e avaliados separadamente.

O módulo CNER reconhece as entidades químicas mencionadas no texto, mapeando cada entidade a um identificador único do ChEBI. Os resultados obtidos passam por um processo de validação que usa semelhança semântica para filtrar erros de reconhecimento. Este módulo é baseado num sistema já existente, tendo sido otimizado para os conjuntos de dados mencionados anteriormente. Estas melhorias consistiram no aumento do número de propriedades exploradas pelo algoritmo de Aprendizagem Automática usado, bem como no melhoramento do processo da validação. Para isto, foi desenvolvido um novo tipo de medida de semelhança semântica, que considera apenas os termos mais relevantes no cálculo da semelhança. O fundamento deste tipo de medida é que ascendentes de um conceito da ontologia com mais relevância serão também os mais importantes para o cálculo. A relevância de um conceito foi estimada através de uma adaptação da medida h-index, usada para avaliar o peso do trabalho publicado por um investigador. Com estas duas melhorias, foi obtida uma medida-F de 82,23% para o conjunto de dados DDI Extraction,

o que representa um aumento de 4,13 pontos percentuais em relação aos resultados obtidos com a versão original do sistema.

O módulo CIE foi desenvolvido para detetar pares de entidades que constituem uma interação, de acordo com o texto, e, se for esse o caso, classificar com um tipo de interação química. Para isto, foram usados algoritmos de Aprendizagem Automática que têm em conta o contexto em que as entidades são mencionadas, as próprias entidades, e informação externa de bases de dados e ontologias. Este módulo foi também avaliado com o conjunto de dados DDI Extraction, obtendo uma medida-F de 74,57% para a deteção de interações, e 65,02% para a classificação de interações. Este resultados são próximos aos obtidos pela melhor participação da competição original.

Os dois módulos foram combinados no sistema IICE, para identificação automática de interações entre entidades químicas. O sistema foi implementado com um interface de linha de comandos, para analisar grandes quantidades de documentos. No entanto, está também disponível numa ferramenta *web*, em `http://www.lasige.di.fc.ul.pt/webtools/iice/`, que permite a qualquer utilizador introduzir um texto para ser analisado pelo sistema. Também é possível introduzir um identificador PubMed para analisar o resumo de um artigo da base de dados MEDLINE. Várias opções foram implementadas na ferramenta, que correspondem a parâmetros descritos nesta dissertação. É possível usar apenas o módulo CIE, caso o texto esteja já anotado com entidades químicas, ou apenas o módulo CNER, para extrair apenas as entidades químicas. O objetivo é o utilizador poder verificar por sim mesmo o efeito dos diferentes parâmetros nos resultados obtidos. É apresentada uma tabela de resumo para as interações químicas identificadas, e outra tabela para os compostos químicos identificados. Como alternativa, os resultados também podem ser descarregados em formato XML.

No domínio biomédico, a extração de interações é uma tarefa ainda com pouco trabalho desenvolvido, quando comparada com outros

domínios e tarefas. Esta dissertação propõe um sistema para extração automática de conhecimento sobre interações químicas de documentos biomédicos. Os resultados obtidos demonstram o potencial deste sistema em aplicações práticas. O uso de técnicas de Aprendizagem Automática permite que este sistema possa ser, no futuro, adaptado a outros tipos de entidades e domínios, usando um conjunto de dados apropriado.

Acknowledgements

First, I would like to thank my supervisor, Professor Francisco Couto, for guiding my work and always being available to help. I would like to thank Fundação para a Ciência e Tecnologia, Professor Francisco Pinto, and the SPnet project for the scholarship that provided me with financial support during the last year. The starting point for this work was possible because of the previous work done by Tiago Grego, who helped me getting started. A big thank you to João Ferreira, who not only read and gave me notes on the papers I wrote, but also presented my work at a conference in Spain. I would like to thank my parents for their unconditional support even though it was not always easy for them to understand exactly what I was doing, and my sister and brother for being great role models. Finally, I would like to thank Diana Galvão, who motivated and inspired me to accomplish my goals.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Everyday, a large amount of biomedical data is generated and made available to the scientific community. This data can be organized in specific data structures, which are easily read by a machine or computer program. However, part of this available biomedical data does not have a defined structure, making it difficult to be processed by a computer program. For example, text, figures and videos often contain biomedical information but those formats are mostly directed to humans, and need a defined process to be transformed into structured data.

One of the major sources of current scientific knowledge is scientific literature, in form of patents, articles or other types of communication. Interactions discovered between chemical compounds are often described in scientific articles (Aronson, 2007). However, the number of documents that a researcher has to retrieve, read and understand to find something useful for his work increases everyday, turning it into a very time-consuming task. Furthermore, the available drug interactions databases are uneven and unable identify correctly the interactions with highest clinical importance (Abarca *et al.*, 2003). One of the biggest sources of biomedical documents is the MEDLINE database (Greenhalgh, 1997), created in 1965. This database contains over 21 million references to journal articles in life sciences, while more than 700,000 were added in 2013. Figure 1.1 shows how this database has increased greatly, storing a lot of knowledge about many topics relevant to biomedicine, including chemical interactions.

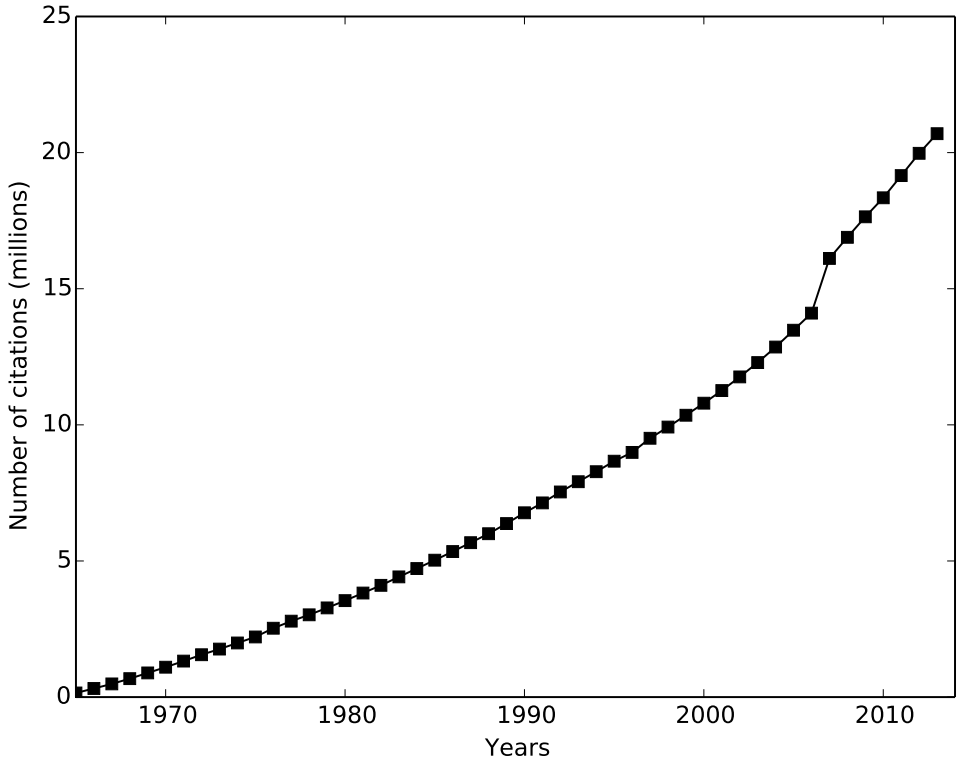

Figure 1.1: Number of citations present in MEDLINE since its beginning in 1950. Data from official statistics available at `www.nlm.nih.gov/bsd/index_stats_comp.html`.

The interactions found in biomedical documents can be used to validate the results of new research or even to find potentially new interactions between two chemical compounds that interact with the same chemical compound. For example, Swanson (1990) found that dietary fish oils might benefit patients with Raynaud's syndrome, by connecting the information present in two different sets of articles that did not cite each other. This inference had been confirmed independently by others in clinical trials (DiGiacomo *et al.*, 1989). In the same study, the author provided two other examples of inferences that could not be drawn from one single article, but only by combining the information of multiple articles. Considering that since that study, the amount of articles available has grown immensely, there are probably many new chemical interactions that can be extracted from this source of information.

Text mining is a research field where techniques are developed to extract useful knowledge from textual data. It has been applied to many domains where information is stored in text documents, for example, news articles, patents, legal cases and scientific papers. Various tasks can be accomplished with Text Mining techniques, for example:

- Named entity recognition (NER) consists in extracting references to relevant entities from text.
- Relation extraction (RE) consists in discovering relations between entities mentioned in the same document.
- Sentiment analysis is used to classify the polarity of a given text relative to an entity or topic.

A more detailed description of these tasks is provided in Chapter 2. As with data mining, there are different approaches that can be applied to perform these tasks. Machine learning approaches have the advantage of being more adaptable than dictionary based approaches, without the manual effort required by rule based approaches. There are various Machine Learning algorithms that can be applied to Text Mining, the most common being Support Vector Machines (Cortes &

Vapnik, 1995) and Condition Random Fields (Lafferty *et al.*, 2001). These algorithms require a text corpus and the expected results for each training document to learn how to extract information from text.

The extraction of interactions between chemical entities requires a first step of identifying the chemical entities mentioned in a given text. This first step is a NER task and it may influence the performance of the Relation Extraction step. Chemical NER is a complex and challenging task, compared to other domains. A single entity maybe be represented by different names, for example, using the systematic nomenclature, molecular formula or brand name. This ambiguity should be resolved by mapping each entity to a universal identifier. Moreover, it is impossible for a single resource to contain every chemical entity that exists, since new chemical compounds are discovered everyday. Dictionary based approaches have limited potential since they cannot identify new entities.

In the simplest case, a chemical interaction consist of two entities, and the relation is symmetrical, i.e., the direction of the relation between the two entities is not relevant. In reality, the relations can be more complex, involving more than two entities, and each entity may have a specific role in the relation.

A chemical interaction is defined in a given text whenever at least two chemical entities are mentioned and at least one of them has some kind of effect on any of the others. Since the focus of this dissertation is on chemical entities with biomedical interest, this effect can be on the chemical structure, concentration value and metabolic pathways of a chemical entity, or other effects relevant to biomedicine.

1.2 Objectives

Biomedical NER has received attention from the community, in the form of research papers, conferences and community challenges. The most advanced systems have obtained good results in the community challenges organized to evaluate the state-of-art, close to the results obtained for domains outside of biomedicine. However, the extraction of chemical interactions has not been researched as much, even though the results obtained with the task can be applied

directly to obtain new knowledge. Fully automated interaction extraction systems are necessary to process the large quantity of text available and extract useful knowledge from it, which can then be applied, for example, to expand databases of chemical compounds and interactions. The objective of this work was to develop a system for extraction of chemical interactions mentioned in biomedical documents, based on Machine Learning and using external resources for validating the results.

Hypothesis: Information about chemical interactions can be efficiently extracted from biomedical documents using Machine Learning techniques and domain knowledge from ontologies.

Machine learning is a subfield of Artificial Intelligence that deals with the design and development of algorithms to perform certain tasks, by learning from example data. The advantage of these algorithms is that they are more flexible than a fixed approach, based on rules or patterns. The results obtained with Machine Learning can then be complemented with domain knowledge.

The system developed should be able to process biomedical documents without any manual annotations, identify the chemical entities mentioned and the chemical interactions described on each document. Each module of this system should then be evaluated using data sets that were created for similar tasks. The Drug-Drug Interactions Extraction task of SemEval 2013 (Segura-Bedmar *et al.*, 2013) provided a corpus of 1025 documents annotated with chemical entities and chemical interactions (Herrero-Zazo *et al.*, 2013). This corpus was used by the participants to evaluate the performance of their systems. The results obtained provided a baseline for the development of other systems.

1.3 Contributions

This work will be fundamentally concerned with proposing a system for automatic extraction of chemical interactions from biomedical documents. This system was divided in two modules, one for the recognition of chemical entities and another for identification of chemical interactions. Thus, the specific contributions can be enumerated as follows:

Chemical Named Entity Recognition (CNER) module: Improvement of the framework developed by Grego & Couto (2013). Since the identification of interactions is dependent on considering the correct entities, it is essential that this module is as optimized as possible. I improved the framework by expanding the feature set and by implementing various validation processes. In particular, I developed a new category of semantic similarity measures which was able to better assess the relevance of concepts, based on the h-index. I used this module to participate on the CHEMDNER task of BioCreative IV. This work resulted in two conference participations and one journal article:

- Lamurias *et al.* (2013). Chemical compound and drug name recognition using CRFs and semantic similarity based on ChEBI. In *BioCreative Challenge Evaluation Workshop*, vol. 2, 75
- Lamurias *et al.* (2014a). Chemical Named Entity Recognition: Improving recall using a comprehensive list of lexical features. In *8th International Conference on Practical Applications of Computational Biology & Bioinformatics (PACBB 2014)*, 253-260, Springer.
- Lamurias *et al.* (2014c). Improving chemical entity recognition through h-index based semantic similarity. *Journal of Cheminformatics* (Minor revisions).

Chemical interactions extraction (CIE) module: A module to classify each pair of chemical entities mention in a given text with a type of interaction, or as not interacting This module is based on Machine Learning techniques, complemented with domain knowledge, and was tested on the DDI Extraction gold standard. The work done for this module resulted in one conference presentation and one journal article .

- Lamurias & Couto (2014). Identifying interactions between chemical entities in text. In *Bioinformatics Open Days, University of Braga.*
- Lamurias *et al.* (2014b) Identifying interactions between chemical entities in biomedical text. *Journal of Integrative Bioinformatics* (In Press).

System for identification of chemical interactions from raw text: I integrated the two modules developed in one system for analyzing biomedical text. This system is accessible via a web tool[1], and was presented on the Lisbon Machine Learning Summer School Demo Day, on Instituto Superior Técnico.

1.4 Overview

The overview of this document is as follows.

Chapter 2 provides an overview of the state-of-art of Text Mining, in particular applied to biomedical documents and chemical entities. The main resources used for this work are presented as well as the ICE framework for chemical entity recognition.

Chapter 3 refers to the improvements that I applied to the ICE framework, in particular the analysis of each newly implemented feature, and the improved semantic similarity measure used for validation.

Chapter 4 deals with the identification of entity pairs that interact in a given sentence. The advantages of kernel methods are discussed, as well as the effect of an ensemble classifier on the classification of interactions.

In Chapter 5 I present the system that I developed for automatic extraction of chemical interactions from raw text.

Finally, on Chapter 6, I discuss the main conclusions of this work, and indicate some directions for future work.

[1] http://www.lasige.di.fc.ul.pt/webtools/iice/

Chapter 2

Related Work

This chapter serves as an overview on the current state of biomedical information extraction with focus on the extraction of chemical interactions from biomedical text. First, the basic concepts necessary to fully understand Text Mining systems based on Machine Learning are presented. A Text Mining system assessment requires specific evaluation measures for the tasks performed, for comparison with other similar systems. Challenge evaluations have been organized, as an effort to compare different approaches to biomedical information extraction. The main evaluation measures and recent challenge evaluations are described in this chapter. Then, the main resources available for biomedical information extraction are presented, including software tools, databases and corpora. While the databases and corpora are focused on the biomedical domain, the software tools can be applied to different domains, assuming the input data is appropriate. In the recent years, the interest for automatic extraction for chemical interactions from text has increased, and as such, chemical interaction extraction systems have been developed and evaluated with domain-specific challenge evaluations. The best systems are reviewed in this chapter. Finally, the approach used by "Identifying Chemical Entities" (ICE) is explained, which was used as a framework for the chemical entity recognition component of this work. This component is required to develop a fully automatic chemical interaction extraction system.

2.1 Text Mining

Text mining consists in extracting useful and relevant knowledge from unstructured text documents (Tan *et al.*, 1999). It can be considered a sub-field of data mining, where the data is in the form of words and sentences. As such, some data mining algorithms can be applied to Text Mining, if the input text is first converted into an appropriate data type, for example, a numeric vector.

While systems developed for news articles have obtained high levels of success, the results are usually lower for the same tasks on scientific text (Dickman, 2003). This is mostly due to the high level of ambiguity within the terms used to refer to entities. Not only the same entity can be mentioned by different nomenclatures or spellings, but the same expression can refer to different entities depending on the context. Furthermore, the sentence structures employed to explain the interactions range from simple to very complex, depending on the mechanism of the interaction and the number of entities.

2.1.1 Main Tasks

The term "Text Mining" is used to describe various tasks with the common goal of extracting useful and relevant information from unstructured text. The actual information that is extracted is what differentiates each task. Different types of information will have different types of applications for the end result. Each task is accomplished using different approaches, which can then be combined to improve the results of another task, or simply to extract more information from the same text. The main Text Mining tasks applied to the biomedical domain will now be described.

2.1.1.1 Document Classification

Document classification is a task with the objective of classifying each document in a set with one or more labels. For example, it may be necessary to classify if each document is relevant to a certain topic, or if it contains information about a certain entity, from a large collection. This can be accomplished by treating the whole document as an instance, and the frequency of each term mentioned as

features, which is usually known as the bag-of-words model. Then, it is possible to apply a supervisioned or semi-supervisioned classification algorithm if there is a set of documents for which the correct labels are known. Otherwise, it is also possible to apply a document clustering algorithm to a set of documents, using the similarity between the feature vectors as a distance measure. This task may be used to assign a topic to each document in a collection, without knowing how the collection is organized.

2.1.1.2 Named Entity Recognition

Named Entity Recognition (NER) consists in classifying the elements in a given text that refer to specific categories. This task usually requires dividing the text in elements, known as tokens, that can then be individually labeled by a classifier. In same cases, the exact location of the entities mentioned may be relevant, while in other cases, it is enough to know that the document mentions a given entity somewhere.

In the biomedical domain, NER systems have been developed to recognize mentions to proteins, genes, cell locations, biological processes, chemical compounds and drugs.

The relevant entities may be constituted by just one word, multiple words in sequence, or multiple words with other words between, each case being more challenging than the other. A common approach that deal with multiple words in sequence is the BIO labels: "Beginning", for the first word of an entity, "Inside", for the other words of the entity, and "Outside", for irrelevant words. To consider entities constituted by words that do not appear sequentially in the sentence, it is necessary to adopt a more complex label system.

The results obtained can be further validated with domain resources such as databases and ontologies. This process is often referred to a normalization, since synonyms are normalized to the same unique identifier. In the biomedical domain, the nomenclature of the entities can vary greatly, which is why an appropriate method should be used to map each entity to the correct identifier.

2.1.1.3 Relation Extraction

The goal of the Relation Extraction task is to identify meaningful semantic relations between the entities mentioned in the text. For this task, it is often implicit that the entities are already identified, manually or with a NER module. A relation may occur between two or more entities, each entity may have its own role in the relation, and it may occur in one specific direction. Furthermore, a relation between a set of entities may be labeled with a specific type. For example, a relation between a gene and a transcription factor may be of the type "activation" or "repression". It is necessary to distinguish between those two types, in order to extract the correct information from the text.

Many types of interactions have been explored, however, in biomedical domain, the focus has been mainly on protein-protein interactions (Krallinger *et al.*, 2008). Other interactions that have been explored are disease-gene, disease-treatment (Bundschus *et al.*, 2008), and drug-drug.

2.1.2 Machine Learning

Machine Learning is a scientific discipline concerned with the design and development of algorithms that allow computers to automatically perform certain tasks, for example, classification of data instances, by learning from training data. The algorithms developed can be applied to a large variety of fields and domains.

Supervised Machine Learning algorithms require a training set, composed by examples of the input data, and the respective expected output. This training set is going to be used to generate a classifier, according to the algorithm chosen. This classifier should be able to classify new unlabeled data, according to the model derived from the training data. It should be noted that the quality and size of the training set will always influence the results produced with a supervised algorithm (Witten & Frank, 2005). A training set should be representative of the data that it is going to be applied to.

Unsupervised learning algorithms do not require the training set to be labeled with the expected output. This is useful if the data labels are unclear or unknown. It is also possible to combine labeled and unlabeled data to train a classifier, with semi-supervised algorithms.

Many Machine Learning algorithms are based on features extracted from the data. These features are inherent to the data, representing properties that distinguish each instance and each label. The selection of the best features for a given task is one of the main challenges in developing a Machine Learning system. The features selected should be specific enough so that the algorithm can learn the difference between the labels, but not too restrict so that it can also be applied to a large variety of data.

The input data for Text Mining is in the form of sentences, paragraphs, documents, or other categories of natural language. For this reason, the input must be converted into a format that is expected for the Machine Learning algorithm.

The bag-of-words model is a common approach to convert textual data into a numeric vector. Some algorithms were already created with text data in mind (Lafferty *et al.*, 2001). In this case, it may be necessary to split the text by word tokens, and generate features for each token. These features are based on the word itself, its context, or external knowledge. The sentence structure can also be used by some algorithms as input data (Zelenko *et al.*, 2003).

The types supervised learning algorithms that are frequently used for Text Mining are now described:

- Decision Trees (Apte *et al.*, 1998): The data is fractioned by branches that represent a condition applied to each instance. The leaves represent the class labels assigned to the instances. This type of algorithm can be applied to text classification, for example.

- Association rules (Wong *et al.*, 1999): Generation of rules according to frequent patterns that occur in the data, generally of the type "if x then y". Useful for extracting relations between entities recognized.

- Naive Bayes (Rennie *et al.*, 2003): The independence of the features is assumed, and a probability model is used to determine the most probable label for each instance. It has been applied to document classification.

- Conditional Random Fields (Lafferty *et al.*, 2001): Labels a sequence of tokens with the most probable sequence of labels, according to the training

data. In this case, the instances are the tokens of a sentence and the context of each token is taken into account.

- Support Vector Machines (Cortes & Vapnik, 1995; Joachims, 1998): The data is represented as points in space, and the algorithm tries to establish a clear division between the instances with the same label. This algorithm can be applied to various types of tasks, as long as the data instances can be represented in a vector space model.

- Kernel-based methods (Zelenko *et al.*, 2003): Class of algorithms that can be applied to Machine Learning, in order to reduce the importance of the feature set. This type of algorithm is based on a kernel function $K : S \times S \rightarrow [0, \infty]$, which is used to express the similarity between two training instances x and y:

$$K(x, y) = \langle f(x), f(y) \rangle$$

where $x, y \in S$ and f is a function that maps an instance to a feature vector, which does not have to be stated explicitly. The kernel function implicitly calculates the dot-product of these feature vectors. This kernel can then be applied to linear Machine Learning algorithm, for example, Support Vector Machines and the Perceptron (Aizerman *et al.*, 1964). With kernel methods, the focus is shifted from feature selection to kernel construction. This is particularly useful for Relation Extraction because the instances are not easily expressed by a feature vector.

Machine learning algorithms usually have parameters that can be changed to optimize the performance. However, caution is necessary when experimenting with different parameters, to prevent overfitting on the example data. Overfitting occurs when the classifier is adjusted to the training data, and memorized various peculiarities of that specific data set, which may not be relevant to other data sets (Dietterich, 1995). Overfitting may also be caused by other reasons, including a limited training set, or poor feature selection. The result is that the classifier seems to perform well for the data available, but when applied to other cases, it has low predictive power.

Overfitting can be avoided by selecting a good evaluation technique. It is common to divide the available data in two or more data sets. One of these partitions is the previously mentioned training set, which should constitute about 70% of the data. Sometimes the training set is only 35% of the data, while the other 35% is the development set, used only to optimize the parameters. The test set is usually about 30% of the data, and it is used to evaluate the system, with the appropriate measures, when it is completed. Each partition should be independent of each other. Another technique to avoid overfitting is cross-validation (Kohavi *et al.*, 1995). This technique consists in dividing the data set in k partitions of equal size, and then testing the classifier on one partition, while training with the rest of the data set. This process is repeated k times and the results of each partition are then evaluated.

2.1.2.1 Other Approaches

While this work is focused on Machine Learning approaches for biomedical Text Mining, it is possible to apply other types of approaches to extract knowledge from text. For a NER task, one common approach is matching the words in the document with a fixed list of entities. This is referred to as dictionary matching (Banville, 2006). This approach usually results in high quality results, which can be easily mapped to a database identifier. However, it is limited, since it cannot recognize a term that is not already contained in the dictionary.

Another common approach involves fixed rules and regular expressions to find entities or interactions. These rules are designed by domain experts, based on language patterns. The results obtained are also of high quality, and it is not as limited as dictionary matching. The main disadvantage of this approach is the time and effort necessary to design the rules, which must be specific for a certain type of text and domain.

Machine learning systems are domain-independent, and more flexible than dictionary and rules-based systems. A pure Machine Learning approach to biomedical information retrieval may not produce results as precise as the other approaches, but it can be enhanced by combining it with a fixed approach. For example, by using matching rules to map the terms recognized with a Machine

Learning classifier to a database identifier, it is possible to filter out some recognition errors made by the classifier (Grego & Couto, 2013).

2.1.3 Natural Language Processing

Natural Language Processing consists in a set of techniques used to derive meaning from raw text written by and for humans. This section is focused on techniques that can process text in some way useful to improve the Text Mining tasks mentioned previously.

2.1.3.1 Tokenization

One of the first processes applied to raw text to be analyzed by a machine is tokenization (Webster & Kit, 1992). Its purpose is to break the text into tokens that can be processed individually and as a sequence. These tokens may consist of simple words, but also of numbers, symbols, phrases and other elements.

The most basic technique for tokenization consists in splitting the text by whitespace and punctuation characters. However, this rule does not always work, and more complex technique should be developed. Usually, a list of abbreviations and acronyms is part of the technique, so that the period at the end is not separated from the letters.

The criteria to what constitutes a token will also vary with the type of text that is going to be processed. In the case of chemical compounds, it may not be desirable to split systematic names, which often contain punctuation and symbols, in more than one token. If this process is not correctly implemented, the performance of a Text Mining system may be limited (Leaman *et al.*, 2008).

2.1.3.2 Stemming

In order to reduce the variability intrinsic to natural language, it is necessary to apply a technique that normalizes variations of the same concept. The objective is to reduce the complexity of the analyzed text by reducing the number of distinct terms used. One of these techniques is stemming, which consists in reducing a word to its stem, or base form. For example, the various forms of a verb should be reduced to the same stem.

Although there are many approaches to this problem, one of the most used is the Porter stemming algorithm (Porter, 1980). This algorithm is based on suffix stripping, and has the advantage of being fast and producing good results, compared to more advanced techniques (Paice, 1996).

Another technique to word normalization is reducing the word to its lemma, called lemmatization. Unlike the Porter algorithm, this technique takes into account the context of the word in the sentence. However, this introduces another source of error to the process, since the sentence structure has to be correctly resolved. Since this technique is more specific than stemming, domain-specific lemmatization tools have been developed (Liu *et al.*, 2012).

2.1.3.3 Part-of-speech tagging

Part-of-speech tagging is often an additional useful source of information for each word in a given sentence. The category of each word depends on both the word itself and its context, since one word may belong to different categories. Approaches developed for news articles and biomedical domain have achieved high performance (Toutanova *et al.*, 2003; Tsuruoka *et al.*, 2005), which is one of the reasons these tags are considered a reliable feature for Text Mining tasks.

2.1.3.4 Parse tree

A parse tree is a representation of the syntactic structure of a sentence. These trees may be constructed according to its constituency grammars, which distinguishes between root, branch and leaf nodes, or according to its dependency grammars, where all nodes are terminal. The output of this process is a structure that can be used as input for another algorithm. The probabilistic methods developed to determine these structures are based on supervised learning techniques (Socher *et al.*, 2013a).

2.1.3.5 Co-reference resolution

To correctly determine the relations between the entities in a given text, it is necessary to resolve co-references to these entities. A co-reference occurs when

two or more expressions refer to the same entity. Usually, one of these expressions in the actual name of the entity, while the others are abbreviated forms, for example, a pronoun or other referring expressions. This is usually one of the last processes applied to a text, since proposed tools require information provided by the processes described previously. One of the currently used solutions for this problem is the Stanford Deterministic Co-reference Resolution System (Lee *et al.*, 2013), which implements a multi-pass sieve co-reference resolution, achieving good results on a shared task dataset. A domain specific solutions for the biomedical domain has also been proposed (Segura-Bedmar *et al.*, 2010).

2.2 Performance Assessment

Methods for evaluating information extraction systems have been developed in order to assess correctly the performance of a system by itself and in comparison to other systems. The evaluation measures developed are used to determine how good a system performs on a given dataset. These measures can be applied to different types of Text Mining. Community challenges are then organized in order to evaluate the state-of-art for a given task and domain. Each team submits the results for a corpus without knowing the expected result, and the organizers compute the evaluation measures for each system. These challenges are essential in order to improve the baseline performance for biomedical Text Mining (Hirschman & Blaschke, 2006). In this section I describe the main evaluation measure used by the community, as well as two recent biomedical Text Mining community challenges.

2.2.1 Evaluation Measures

The performance of an information extraction system is evaluated by testing it with an unlabeled corpus. Although the corpus has been previously annotated with the expected results by domain experts, the system should not use these annotations to generate results. A gold standard is an annotated corpus used to evaluate information extraction systems, and its format depends on the task being evaluated. For NER, it may be a corpus annotated with the position of

every entity mentioned on each document, or just a list of every entity mention on each document. For a Relation Extraction task, the gold standard should be list of pairs of entities that are interacting on each document. In order to evaluate this task separately from the entity recognition task, a list of all entities mentioned on each document should also be provided.

For a given information extraction task, it should be defined what is considered a positive result. In the case of entity recognition, a positive result is an entity identified in the text, while for Relation Extraction, it is an interaction found between two entities in the text. Likewise, a negative result is a piece of text that was not identified as a relevant entity, or a pair of entities that was not classified as an interaction, respectively.

The positive results identified by a system that are actually correct according to the gold standard are known as True Positives (TP). The ones that were incorrectly identified as positive are known as False Positives (FP). The same logic applies to True Negatives (TN) and False Negatives (TN). These four possible types of result of a gold standard evaluation can be represented in a contingency table, as in Table 2.1.

Table 2.1: Contingency table for the types of result obtained with a Text Mining system.

	Gold Standard Positive	Gold Standard Negative
Positive outcome	True Positives (TP)	False Positives (FP)
Negative outcome	False Negatives (FN)	True Negatives (TN)

The objective of an information retrieval system is to maximize the number of TP and TN, and minimizing the number of FP and FN. However, to compare the results obtained within different data sets, relative measures are calculated from the values on Table 2.1, since the maximum number of TP and TN will vary with the data set. Precision is the fraction of positive results that were correctly classified:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{2.1}$$

This measure represents how often the results obtained with the system are correct. Systems with high precision values are unlikely to extract incorrect information. Recall is a measure of how many positive results were extracted by the system, relative to the total for that gold standard:

$$\text{Recall} = \frac{TP}{TP + FN} \tag{2.2}$$

A system that has obtained a recall of 100% for a given gold standard has extracted all the relevant information. However it may have also extracted information that was incorrect or irrelevant, which means that the number of FP may be higher than 0, and the precision value would be less than 100%. Likewise, a system may identify only correct information, but just a fraction of what it was supposed to identify, according to the gold standard. While these two measures have a defined meaning in the context of information extraction, it is often useful to combine them in order to express the performance level by just one number. The F-measure is the harmonic mean of precision and recall and it is often used to determine the best system on a community challenge:

$$\text{F-measure} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{2.3}$$

To achieve a high F-measure, it is necessary to obtain both high precision and recall values.

It should be noted that these measures depend not only on the performance of the system but also on the quality of the manual annotations of the gold standard. The inter-annotator agreement estimates the quality of an annotated corpus and it is calculated with the kappa coefficient:

$$k = \frac{P(A) - P(E)}{1 - P(E)} \tag{2.4}$$

where $P(A)$ is percentage of times the annotators agreed and $P(E)$ is the percentage of times it was expected for them to agree by chance (Carletta, 1996). A k of 100% would indicate that the annotators agreed on every annotation.

2.2.2 Community Evaluations

2.2.2.1 CHEMDNER task

The CHEMDNER task of BioCreative IV consisted in the identification of named chemical entities from PubMed abstracts (Krallinger *et al.*, 2014b). There were two types of predictions the participants could submit for the CHEMDNER task: a ranked list of unique chemical entities described on each document, for the Chemical Document Indexing (CDI) subtask, and the start and end indices of each chemical entity mentioned on each document for the Chemical Entity Mention (CEM) subtask. Using the CEM predictions, it was possible to generate results for the CDI subtask, by excluding multiple mentions of the same entity in each document. A gold standard for both subtasks was available to the participants, which could be used evaluate the performance of each approach, with the evaluation script released by the organization. Each team was allowed to submit up to five different runs for each subtask.

Since BioCreative is nowadays a reference in biomedical Text Mining evaluations, there was much interest in this task, with 27 teams participating on at least one subtask. The organization estimated that a dictionary based approach, using only the entities annotated on training and development sets, would obtained a F-measure of 53.85% for the CDI task and 57.11% for the CEM task. The best team achieved a F-measure of 88.20% for the CDI task and 87.39% for the CEM task. Most teams used Machine Learning techniques and external domain lexical resources to develop their systems.

2.2.2.2 DDI Extraction task

The DDI Extraction was part of 2013 edition of SemEval, a series of workshops on semantic evaluation (Segura-Bedmar *et al.*, 2013). This was the second edition of this task, which was composed by two subtasks. The first subtask consisted in the recognition and classification of pharmacological substances mentioned in biomedical texts, while the second consisted in the identification and classification of drug-drug interactions, also from biomedical texts. Each team could submit results for just one of the tasks, since the test sets were independent. However, the train set was common to both subtasks, each document being annotated

with drug entities and drug-drug interactions. The corpus released for this task consisted of MEDLINE abstracts and descriptions of drug-drug interactions from DrugBank.

A total of 6 teams participated in the first subtask, while 8 teams submitted results for the second subtask. The best team achieved a F-measure of 71.5% for the NER task and 65.1% for the Relation Extraction task. However, considering only the documents from DrugBank, the results obtained were much better, with the best F-measure being 87.8% and 67.6% for each of the two subtasks, respectively.

2.3 Resources

The following sub-sections aim to describe the main resources for biomedical information extraction. The Text Mining algorithms and natural language processing techniques previously described have been implemented in software packages, which can then be applied to any compatible data. To achieve high performance, information extraction systems can combine various tools to process the input text. This sections describes the main tools that can be used to build and information extraction system. The Machine Learning classifiers should be trained with an appropriate corpus. For Relation Extraction, the corpus has to be annotated with the relevant entities, and the interacting entities should be identified. In this section, I will describe two corpora that have been released recently, annotated with chemical entities and chemical interactions. These corpora are essential for development and evaluation of chemical interaction extraction systems. Finally, I will present some of the most popular sources of biomedical and chemical information.

2.3.1 Machine Learning

2.3.1.1 Natural Language Processing

Fortunately, there are various tools available, which can process text and perform natural language processing tasks on it. These tools can then be combined as a pre-processing step for a Text Mining system.

The Natural Language Toolkit (NLTK) (Bird *et al.*, 2009) is a platform for natural language processing in Python, that can be used for sentence splitting, tokenization, POS tagging, stemming, lemmatization and manipulation of parse trees. It incorporates models based on various corpora, mostly from the news domain, but also some biomedical corpora, for example, the BioCreAtIvE-PPI corpus, for protein-protein interactions.

The Stanford Natural Language Processing Group has released a set of tools, which are integrated together in the CoreNLP suite (Manning *et al.*, 2014). This suite provides a tool for tokenization, lemmatization, POS tagging (Toutanova & Manning, 2000), dependency parsing (Klein & Manning, 2003), co-reference resolution (Lee *et al.*, 2013) and Named Entity Recognition of specific categories, including numeric entities (Finkel *et al.*, 2005). While the models used by each tools are trained with news articles, they can also be applied to biomedical texts.

The BLLIP reranking parser (also known as Charniak-Johnson parser) (Charniak, 2000) is a constituency parser and discriminative maximum entropy reranker, used to determine the parse tree from sentences. There is a self-trained reranking model augmented by biomedical texts that is available for this tool (McClosky & Adviser-Charniak, 2010). As expected, this model provides better results with biomedical texts than the Stanford parser (Segura-Bedmar *et al.*, 2014).

2.3.1.2 Machine Learning tools

Machine learning toolkits are used to test and compare the results obtained with various algorithms. Scikit-learn (Pedregosa *et al.*, 2011) is a Python-based general purpose toolkit for Machine Learning. It provides implementations of many algorithms, as well as other common functions, for example, feature extraction, parameter optimization, and cross-validation. Weka (Hall *et al.*, 2009) is another general purpose toolkit for Machine Learning, available in a Java API, Java class, and graphical user interface. It also provides some common functions, for preprocessing the input data and model evaluation.

In some cases, it may be more practical to use an algorithm-specific tool. The Machine Learning algorithms previously described have been implemented by various tools, which can differ in the performance and default parameters. One

of the most used Conditional Random Fields implementations is Mallet (McCallum, 2002), which is Java-based and also performs other Text Mining tasks, for example, document classification and clustering. Other implementations exist, for example, CRFsuite (Okazaki, 2007), which has been reported to be much faster than Mallet.

The most popular Support Vector Machines implementations are SVM-light (Joachims, 1999) and LIBSVM (Chang & Lin, 2011). The SVM-light-TK (Joachims, 1999; Moschitti, 2006) is an implementation of the SubSet Tree kernel, based on SVM-light. It uses the parse tree of a sentence to identify pairs of interacting entities. The jSRE tool implements a non-linear kernel, the Shallow Language kernel (Giuliano *et al.*, 2006), for classification of pairs of entities. This tool is based on LIBSVM and has been applied to the biomedical domain, obtaining good results (Segura-Bedmar *et al.*, 2011). The Shallow Language kernel takes into account both the global and local context of each entity to determine if they are interacting or not.

2.3.2 Corpora

Recently, some community challenges have focused on identification of chemical entities and chemical interactions form biomedical text. These challenges provide a corpus for training and evaluation of the competing systems. The objective of this section is to describe the corpora released for the Drug-Drug Interaction Extraction task of SemEval 2013 and for the CHEMDNER task of BioCreative IV. The results obtained with these gold standards can then be compared with those obtained by the teams that participated in each competition.

However, to be fair, evaluations done outside of the scope of the competition are not completely comparable with the participating teams since their work was limited by the submission deadline.

2.3.2.1 CHEMDNER corpus

The CHEMDNER corpus consists of 10,000 MEDLINE titles and abstracts and was partitioned randomly in three sets by the authors: training, development and test (Krallinger *et al.*, 2014a). The chosen articles were sampled from a list of

articles published in 2013 by the top 100 journals of a list of categories related to the chemistry field. These articles were manually annotated by a team of curators with background in chemistry. Each annotation consisted of the article identifier, type of text (title or abstract), start and end indices, the text string and the type of chemical entity, which could be one of the following: "trivial", "formula", "systematic", "abbreviation", "family" and "multiple". There was no limit for the number of words that could refer to a CEM but due to the annotation format, the sequence of words had to be continuous. There were a total of 59,004 annotations on the training and development sets, which consisted of 7,000 documents. The test set consisted of 3,000 documents and was annotated with 25,351 chemical entities. Figure 2.1 provides an example of the format of this corpus.

Abstract

```
23194825    Neurotoxicity of "ecstasy" and its metabolites in human dopaminergic differentiated SH-SY5Y
cells.  "Ecstasy" (3,4-methylenedioxymethamphetamine or MDMA) is a widely abused recreational drug,
reported to produce neurotoxic effects, both in laboratory animals and in humans. MDMA metabolites can
be major contributors for MDMA neurotoxicity. This work studied the neurotoxicity of MDMA and its
catechol metabolites, α-methyldopamine (α-MeDA) and N-methyl-α-methyldopamine (N-Me-α-MeDA) in human
dopaminergic SH-SY5Y cells differentiated with retinoic acid and 12-O-tetradecanoyl-phorbol-13-acetate.
Differentiation led to SH-SY5Y neurons with higher ability to accumulate dopamine and higher resistance
towards dopamine neurotoxicity. MDMA catechol metabolites were neurotoxic to SH-SY5Y neurons, leading to
caspase 3-independent cell death in a concentration- and time-dependent manner. MDMA did not show a
concentration- and time-dependent death. Pre-treatment with the antioxidant and glutathione precursor,
N-acetylcysteine (NAC), resulted in strong protection against the MDMA metabolites' neurotoxicity.
```

Annotations

```
23194825    A   345 370 N-methyl-α-methyldopamine   SYSTEMATIC
23194825    A   372 383 N-Me-α-MeDA ABBREVIATION
23194825    A   441 454 retinoic acid   TRIVIAL
23194825    A   459 496 12-O-tetradecanoyl-phorbol-13-acetate   SYSTEMATIC
23194825    A   48  52  MDMA    ABBREVIATION
23194825    A   571 579 dopamine    TRIVIAL
23194825    A   610 618 dopamine    TRIVIAL
23194825    A   634 638 MDMA    ABBREVIATION
23194825    A   639 647 catechol    TRIVIAL
23194825    A   787 791 MDMA    ABBREVIATION
23194825    A   887 898 glutathione TRIVIAL
23194825    A   910 926 N-acetylcysteine    SYSTEMATIC
23194825    A   928 931 NAC ABBREVIATION
23194825    A   976 980 MDMA    ABBREVIATION
23194825    T   18  25  ecstasy TRIVIAL
```

Figure 2.1: Example of the text and annotations provided by the CHEMDNER corpus. The Abstract section consists of the PMID, title and abstract text, separated by tabs. The Annotations section consists of PMID, Title (T) or Abstract (A), start index, end index, text string and type of chemical entity, also tab separated.

The inter-annotator agreement estimated for this corpus was 91% when considering only the matching of the entities, and 85.26% when also taking into

account the types of chemical entities.

2.3.2.2 DDI corpus

The DDI corpus was originally released for task 9 of SemEval 2013, which consisted in extracting drug-drug interaction from biomedical texts (Herrero-Zazo *et al.*, 2013). This corpus is composed by 792 texts from the DrugBank database and 233 MEDLINE abstracts, and was partitioned in two sets by the authors: train and test. Each document is annotated with drug names and drug-drug interactions. The types of interactions considered by this corpus were: "mechanism", "effect", "advice" or "int" when none of the others was applicable. Table 2.2 provides an example of each type of interaction from the corpus.

Table 2.2: Examples of interactions from the DDI corpus. The entities that constitute the interaction are highlighted.

DDI type	Sentence
advise	Administration of a higher dose of **indinavir** should be considered when coadministering with **megestrol acetate**.
effect	When administered concomitantly with **ProAmatine**, **cardiac glycosides** may enhance or precipitate bradycardia, A.V.
mechanism	In vivo, the plasma clearance of **ropivacaine** was reduced by 70% during coadministration of **fluvoxamine** (25 mg bid for 2 days), a selective and potent CYP1A2 inhibitor.
int	**Trilostane** may interact with **aminoglutethimide** or mitotane (causing too great a decrease in adrenal function).

There was a total of 18,502 chemical entities and 5,028 interactions in this dataset. The estimated inter-annotator agreement for the relation of this corpus was of 83.85% for the DrugBank documents and 62.13% for the MedLine documents.

2.3.3 Databases and Ontologies

Several efforts have been made in order to develop open accessible repositories of biomedical knowledge. An ontology is a data structure used to represent concepts within a domain and their relationships (Gruber, 1993). With an ontology, it is

possible to compare the terms using the structural component of the ontology. This section describes three popular information resources for chemical entities, which can be used to validate the results obtained with a biomedical information extraction system.

2.3.3.1 Chemical Entities of Biological Interest

Chemical Entities of Biological Interest (ChEBI) is a freely available database and ontology of small molecular entities with biological interest, containing more than 40,000 entries (Hastings *et al.*, 2013). The ontology is a Directed Acyclic Graph (DAG), which means that each concept can have multiple ancestors. It is composed by three sub-ontologies: "chemical entity", "role" and "subatomic particle", while nine different types of relationships are considered. Since a recent update, all database entries have a "is a" relationship within the ontology, which means that the ontology now has as many concepts as the database.

2.3.3.2 Gene Ontology

The objective of Gene Ontology is to develop a dynamic, controlled vocabulary that is able the adapt with the high rate at which biomedical knowledge is produced (Ashburner *et al.*, 2000). This project has been very successful, and has been applied to many bioinformatics projects. The ontology itself is composed by three sub-ontologies: "biological process", "molecular function" and "cellular component", and three types of relations are considered: "is a", "part of" and "regulates".

Recently, GO developers have worked closely with ChEBI developers in order to align the chemical concepts present in the GO with the respective concept in the ChEBI ontology (Consortium *et al.*, 2012). This means that two chemical entities that exist in both ontologies may be compared differently on each one.

2.3.3.3 DrugBank

The DrugBank database is a resource for detailed biochemical and pharmacological information about drugs and their mechanisms, including interactions with other drugs (Law *et al.*, 2014). Its latest version contains 7,677 drug entries,

and it is available to the public as a single file that can be downloaded from the homepage.

2.4 State-of-the-art of Chemical Interaction Extraction

In this section, I will cover the state-of-the-art approaches for identification of chemical interactions, based on recent community challenges. In the last few years, chemical entity recognition systems have switched from dictionary based approaches to Machine Learning techniques, mostly Conditional Random Fields and Support Vector Machines, which led to great improvements in the results obtained. For this reason, there are many systems that perform recognition of chemical terms in text. However, only a fraction of these systems also extract the chemical interactions described in the same text. The interest of the community in this type of task has grown over the years, and the results have also been improving. Although protein-protein interactions are usually the main case study for extraction of interactions from biomedical texts, chemical interactions have also received some attention from the community. For instance, the best F-measure for the detection of interactions task improved from 65.74% to 80% between the 2011 and 2013 editions of the DDI extraction task (Segura-Bedmar *et al.*, 2013). The best systems for this type of task employ Machine Learning algorithms, in particular non-linear kernel SVMs and biomedical language models to identify interactions described in the text. Table 2.3 summarizes the main approaches and resources used by each system.

Table 2.3: Summary of the state-of-the-art systems to extraction of chemical interactions from text.

System	Main Approaches	External resources	Interactions
HyRex	Hybrid kernel SVMs	SVM-Light-TK, jSRE	DDI
TEES 2.0	SVM	WordNet, DrugBank	Various
WBI	Kernel-based methods	DrugBank, Phare Ontology, TEES, jSRE	DDI

HyREX is a system for detection and classification of drug-drug interactions (Chowdhury & Lavelli, 2013a). The main feature of this system is that it exploits the scope of negation of a sentence to reduce the number of candidate pairs. This is applied on the first of two stages that constitute this system. In the second stage, a hybrid kernel is used to classify each pair with a label, corresponding to a type of interaction, or none. This system obtained the best performance on the DDI Extraction task of SemEval 2013. The source code for this system is available at `https://github.com/fmchowdhury/HyREX`.

The Turku Event Extraction System (TEES) is a system that performs recognition of chemical entities and chemical interactions, besides other types of relations and events, from biomedical texts (Björne *et al.*, 2011). This system is based on SVM classifiers trained with deep syntactic features and information from external resources, achieving good results on various community challenges, including both editions of the DDI extraction task. The source code for this system is available at `https://github.com/jbjorne/TEES/`.

Thomas *et al.* (2013) have combined several kernel-based methods to identify and classify drug-drug interactions. Furthermore, they also employ TEES, DrugBank and the Phare Ontology (Coulet *et al.*, 2011) as external sources of information. This approach achieved the best performance of the 2011 DDI extraction task and second best performance of the 2013 DDI Extraction task.

2.5 ICE framework

"Identifying Chemical Entities" (Grego & Couto, 2013) (ICE) is framework for chemical entity recognition that was adapted for this work. This framework was originally developed with a corpus of forty patent documents, manually annotated with ChEBI terms by a team of curators from ChEBI and the European Patent Office. The main components of this framework will now be described.

2.5.1 CRF entity recognition

The ICE framework is based on the Conditional Random Fields (CRF) implementations of Mallet, with the default values. In particular, only an order of 1

is used for the CRF algorithm. The following features are extracted from the training data to train the classifiers:

Stem: Stem of the word token with the Porter stemming algorithm

Prefix and Suffix size 3: The first and last three characters of a word token.

Number: Boolean that indicates if the token contains digits.

Furthermore, each token is given different labels depending on whether it was not a chemical entity, a single word chemical entity, or the start, middle or end of a chemical entity (Grego *et al.*, 2009). Table 2.4 provides an example of the features generated for a fragment of text, as well as the labels.

Table 2.4: Example of a sequence of the ICE features, and the corresponding label, derived from a sentence fragment (PMID 23194825). The "Number" feature is omitted since none of the tokens were numbers.

Token	Prefix 3	Suffix 3	Stem	Label
Cells	Cel	lls	Cells	Not Chemical
exposed	exp	sed	expos	Not Chemical
to	to	to	to	Not Chemical
α-MeDA	α-M	eDA	α-MeDA	Chemical
showed	sho	wed	show	Not Chemical
an	an	an	an	Not Chemical
increase	inc	ase	inscres	Not Chemical
in	in	in	in	Not Chemical
intracellular	int	lar	intracellular	Not Chemical
glutathione	glu	one	glutathion	Chemical
(	(	(	(	Not Chemical
GSH	GSH	GSH	GSH	Chemical
)	)	)	)	Not Chemical
levels	lev	els	level	Not Chemical

Since Mallet does not provide a confidence score for each label, the source code was adapted, so that for each label, a probability value is also returned, according to the features of that token. This information is used to adjust the precision of the predictions obtained, and to rank them according to how confident the system is about the extracted mentions being correct.

2.5.2 ChEBI resolution

After having recognized the named chemical entities, this framework resolves each term to the ChEBI ontology. The resolution method takes as input the string identified as being a chemical compound name and returns the most relevant ChEBI concept along with a mapping score (Grego *et al.*, 2012).

To perform the search for the most likely concept for a given input string, an adaptation of FiGO, a lexical similarity method (Couto *et al.*, 2005), is employed. This adaptation compares the constituent words in the input string with the constituent words of each concept, to which different weights have been assigned according to its frequency in the ontology vocabulary. A mapping score between 0 and 1 is provided with the mapping, which corresponds to a maximum value in the case of a concept that has the exact same name as the input string.

2.5.3 ChEBI Semantic Similarity

The calculation of the semantic similarity between two concepts is based on the ChEBI ontology:

$$\text{sim}(c_1, c_2) = n, n \in [0, 1] \wedge c_1, c_2 \in \text{ChEBI}$$

Three measures are implemented for the ChEBI ontology: Resnik, simUI and simGIC. Then, the semantic similarity is calculated between one concept and every other concept recognized in the same text window. The maximum value returned by this method for each recognized concept is used as the semantic similarity score. This means that if a recognized concept has a high similarity value with at least one other concept in the same text window, it will also have a high semantic similarity score (Grego & Couto, 2013). The assumption is that if two entities are mention in the same text window, they should share some semantic similarity and are more likely to be correct.

2.5.4 Post-processing

Some simple rules are implemented, in an effort to improve the quality of the annotations:

1. Exclude if one of the words is in a stop words list
2. Exclude text with no alphanumeric characters
3. Delete the last character if it is a dash ("-")

A list of common English words is used as stop words in post-processing. If a recognized chemical entity is part of this list or one of the words on the list is part of the chemical entity, then it is considered a recognition error and it it not considered a chemical entity.

Chapter 3

Chemical Named Entity Recognition

A required first step for the automatic identification of chemical interactions is the recognition of chemical entities mentioned in a given text. As a starting point, I adapted the ICE framework, by improving the results obtained for the CHEMDNER corpus. Then, I evaluated these improvements with the DDI corpus. The objective of this chapter was to optimize the CNER module as much as possible, so that it would not limit the performance of the CIE module. Even though the CNER module initially achieved high precision values, the recall was not as high. If some chemical entities are not considered for the CIE module, it will not identify the interactions that involve those entities. As such, the goal was to improve the recall, with minimal effect on the precision, which would also improve the F-measure.

3.1 Methods

Since the patent corpus initially used on ICE was small and not used by other similar systems, new classifiers were trained for the DDI and CHEMDNER corpus. For each type of chemical entities considered on each of these corpus, one additional training dataset was generated and type-specific classifier was trained with it. Each input document is classified with this set of classifiers, and the results are merged. The objective of this strategy was to recognize more entities

than a general classifier would not recognize. However, I did not attempt to classify each recognized entity with a type, since each entity is mapped to a ChEBI identifier, which should provide more domain-specific information.

3.1.1 Validation process

The output provided for each putative chemical named entity recognized is the Conditional Random Fields (CRF) classifier's confidence score, the ChEBI mapping score and the most similar putative chemical named entity mentioned on the same document through the maximum semantic similarity score. The features set for each prediction was composed by these three scores. When a chemical entity mention is detected by at least one classifier, but not all, the confidence score for the classifiers that did not detect this mention was considered to be 0. These features were used to train a classifier to filter false positives from the results, with minimal effect on the recall value. The predictions obtained by cross-validation on the CHEMDNER training and development sets were used to train different classifiers with Weka, using the different learning algorithms implemented by the toolkit. The best results were obtained with the Random Forests ensemble learning approach.

I then experimented with different combinations of training corpora and validation approaches to evaluate the performance of the module on the CHEMDNER corpus. Each of these combinations corresponds to a testing run submitted for the CHEMDNER task of BioCreative IV.

Different runs use different corpora for the CRF step: each uses (1) either the CHEMDNER corpus by itself or (2) the CHEMDNER corpus along with the DDI and patents (PAT) corpora. DDI and PAT were not annotated with the same criteria used for the CHEMDNER corpus, and do not contain the same type of texts. The DDI corpus is focused on drug names and contains drug interaction descriptions and PubMed abstracts, while PAT contains only patents annotated with chemical named entities.

To validate the CRF results, I employed three different approaches: (1) The first approach was to map the recognized entities to ChEBI and then apply the

Semantic Similarity Measure (SSM) described on Section 2.5.3 to filter the entities based on a fixed threshold. (2) The second approach was to combine the confidence scores obtained with Mallet and ChEBI mapping score with the SSM values for each entity, computing a new score which was also used to filter the CRF results based on a threshold (COMBINED). (3) Finally, I used the three scores independently to produce a Random Forests classifier to classify each entity as a true positive or a false positive (RF).

Experimenting with cross-validation on the training and development sets, I assembled different combinations of these approaches (see Table 3.1).

Table 3.1: Corpora and validation approaches used for each testing run.

	Corpora		Validation		
Run	CHEMDNER	DDI/PAT	SSM	COMBINED	RF
1	X	X			X
2	X			X	
3	X	X			
3*	X				
4	X		X		
5	X	X	X		

On run 1, I used the full set of corpora alongside the RF validation. This was decided after noticing that the Random Forest classifiers provided a better balance between precision and recall than a simple approach based on a score and threshold (approaches SSM and COMBINED).

For run 2, I used only the CHEMDNER corpus and the COMBINED validation process, since the combined score of each entity is more detailed than just one of the values. I determined empirically the threshold of 0.8 for this run, which gave the maximum precision value for the module.

Run 3 is equivalent to a baseline for the validation processes. In fact, this run uses only the results obtained with a CRF classifier trained with the full set of corpora, without a validation step. To better understand the effect of the training corpus, I also created a run 3*, where the CRF was trained with the CHEMDNER corpus only. The results of these two runs (3 and 3*) establish the maximum recall value that can be expected with the CNER module, as they result in a non-filtered list which the validation step trims down. The perfect validation step should be

able to remove from the CRF results all the false positive recognitions, but can never increase the number of correctly recognized entities. Notice that run 3* was not submitted for evaluation at the CHEMDNER task, as only 5 runs were allowed per team.

Runs 4 and 5 use the SSM validation step, along with either the CHEMDNER corpus alone (run 4) or the full set of corpora (run 5) This selection was done in order to evaluate the performance of the SSM validation approach, since it had been applied before to a different gold standard with success. The threshold value applied (0.4) was based on the experiments done by Grego & Couto (2013).

According to the corpora used, run 3 should be used as a baseline for runs 1 and 5, while run 3* should be used as a baseline for runs 2 and 4.

3.1.2 Expanded feature set

After participating on the CHEMDNER challenge with the runs previously described, I further improved two aspects of this approach, with the objective of improving the recall, without affecting the precision. As such, thirteen new features were integrated on the CNER module, based on orthographic and morphological properties of the words used to represent the entity, and inspired by other CRF-based chemical NER systems (Batista-Navarro *et al.*, 2013; Campos *et al.*, 2013; Huber *et al.*, 2013; Leaman *et al.*, 2013; Usié *et al.*, 2013). I studied the effect of adding one new feature at a time, while always keeping the four original features constant. The following features were integrated:

- **Prefix and Suffix sizes 1, 2 and 4:** The first and last n characters of a word token.
- **Greek symbol:** Boolean that indicates if the token contains Greek symbols.
- **Non-alphanumeric character:** Boolean that indicates if the token contains non-alphanumeric symbols.
- **Case pattern:** "Lower" if all characters are lower case, "Upper" if all characters are upper case, "Title" if only the first character is upper case and "Mixed" if none of the others apply.

- **Word shape:** Normalized form of the token by replacing every number with '0', every letter with 'A' or 'a' and every other character with 'x'.
- **Simple word shape:** Simplified version of the word shape feature where consecutive symbols of the same kind are merged.
- **Periodic Table element:** Boolean that indicates if the token matches a periodic table symbols or name.
- **Amino acid:** Boolean that indicates if the token matches a 3 letter code amino acids.

For example, for the sentence fragment "Cells exposed to α-MeDA showed an increase in intracellular glutathione (GSH) levels", the list of tokens obtained by the tokenizer and some possible features are shown on Table 3.2.

Table 3.2: Example of a sequence of some the new features, and the corresponding label, derived from a sentence fragment (PMID 23194825).

Token	Prefix 4	Suffix 4	Case pattern	Word shape	Label
Cells	Cell	ells	titlecase	Aaaaa	Not Chemical
exposed	expo	osed	lowercase	aaaaaaa	Not Chemical
to	to	to	lowercase	aa	Not Chemical
α-MeDA	α-Me	MeDA	mixed	xxAaAA	Chemical
showed	show	owed	lowercase	aaaaaa	Not Chemical
an	an	an	lowercase	aa	Not Chemical
increase	incr	ease	lowercase	aaaaaaaa	Not Chemical
in	in	in	lowercase	aa	Not Chemical
intracellular	intr	ular	lowercase	aaaaaaaaaaaaa	Not Chemical
glutathione	glut	ione	lowercase	aaaaaaaaaaa	Chemical
(	(	(	-	x	Not Chemical
GSH	GSH	GSH	uppercase	AAA	Chemical
)	)	)	-	x	Not Chemical
levels	leve	vels	lowercase	aaaaaa	Not Chemical

After applying the validation process SSM previously described for each new feature, I was able to compare the effect of each one on the results. This validation process was chosen since it was shown to achieve a good compromise between precision and recall. However, the threshold was set at 0.8, which results in very

high precision and low recall. My objective was to improve the recall for high precision levels. Then, I selected the features that achieved higher precision, recall and F-measure for that threshold, creating three sets of features for each metric and a fourth set with all the features tested, for comparison.

3.1.3 Improved validation process

I used the maximum semantic similarity value of each predicted chemical entity to the other entities identified in the same fragment of text to filter entities incorrectly predicted by the CRF classifiers.

The simUI measure (Gentleman, 2005) is an edge-based approach to measure the semantic similarity between two classes. Given two classes c_1 and c_2, and the set of their ancestors $asc(c_1)$ and $asc(c_2)$, this measure is equal to the number of classes in the intersection between $asc(c_1)$ and $asc(c_2)$ divided by the number of classes in the union of the same two sets:

$$\text{simUI}(c_1, c_2) = \frac{\#\{t \mid t \in asc(c_1) \cap asc(c_2)\}}{\#\{t \mid t \in asc(c_1) \cup asc(c_2)\}}$$

A similar approach for measuring semantic similarity is the simGIC measure (Pesquita *et al.*, 2007). In this case, each ancestor is weighted by its information content (IC), which is a measure of the specificity of a concept. The simGIC is defined as the sum of the IC of the classes in the intersection between $asc(c_1)$ and $asc(c_2)$ divided by the sum of the IC of the classes in the union of the same two sets:

$$\text{simGIC}(c_1, c_2) = \frac{\sum\{\text{IC}(t) \mid t \in asc(c_1) \cap asc(c_2)\}}{\sum\{\text{IC}(t) \mid t \in asc(c_1) \cup asc(c_2)\}}$$

The hierarchical structure of the ontology can be used to quantify the IC of each class. Seco *et al.* (2004) proposed an intrinsic IC as a function of the number of sub-classes and the maximum number of classes in the ontology:

$$\text{IC}(c) = 1 - \frac{\log(\text{sub-classes}(c) + 1)}{\log(C)}$$

where sub-classes(c) is the number of sub-classes of c and C is the total number of classes in the ontology.

Both simUI and simGIC consider every ancestor up to the root. These measures could be improved by selecting only the ancestors that are more relevant in the ontology. I estimated the relevance of a class by adapting the h-index (Hirsch, 2005) to the ChEBI ontology, defining it as follows: A term has index h if h of its Np children have at least h children each and the other $(Np - h)$ children have $\leq h$ children each. Figure 3.1 shows an example of a ChEBI entity (CHEBI:24346) with an h-index of 2. Classes that are leaf nodes or classes that have only leaf nodes as sub-classes have an h-index of 0.

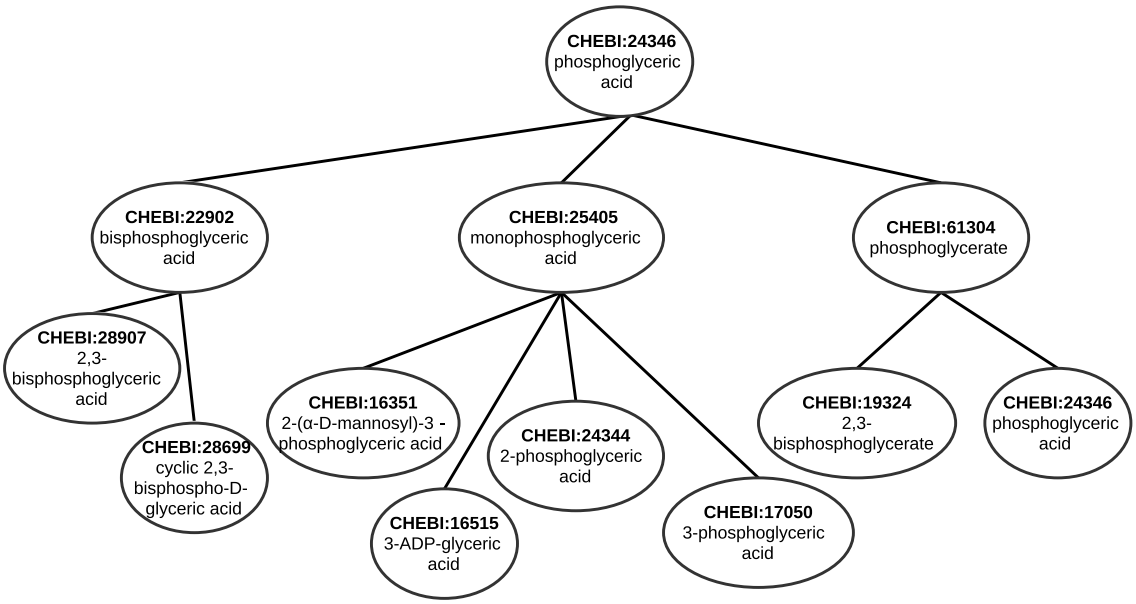

Figure 3.1: Section of the ChEBI ontology showing a term (CHEBI:24346) with a h-index of 2, since 2 of its child nodes have at least 2 other child nodes, and the other child node has no more than 2 child nodes.

Then, I adapted the simUI and simGIC measures to exclude ancestors with an h-index lower than a certain threshold α. Only the ancestors with h-index higher or equal to α are considered for $asc(c_1)$ and $asc(c_2)$.

$$\text{simUI}_h(c_1, c_2) = \frac{\#\{t \mid t \in asc(c_1) \cap asc(c_2) \wedge \text{h-index}(t) \geq \alpha\}}{\#\{t \mid t \in asc(c_1) \cup asc(c_2) \wedge \text{h-index}(t) \geq \alpha\}}$$

$$\text{simGIC}_h(c_1, c_2) = \frac{\sum\{\text{IC}(t) \mid t \in asc(c_1) \cap asc(c_2) \wedge \text{h-index}(t) \geq \alpha\}}{\sum\{\text{IC}(t) \mid t \in asc(c_1) \cup asc(c_2) \wedge \text{h-index}(t) \geq \alpha\}}$$

Using lower α values, fewer ancestors are excluded and consequentially, the similarity values should be closer to the ones obtained with the original measures. As the threshold α is increased, only the most relevant classes are considered and the semantic similarity values deviate more from the original.

I performed a similar recognition process to what was used previously on the framework, but now using the simUI and simGIC similarity measures, and the adapted versions based on h-index filtering.

My objective was to improve the overall recall while maintaining high precision values, by better filtering out false positives from the results obtained with the CNER module. Using my adapted versions of the simUI and simGIC measures, I expected more false positives to be removed, for the same number of true positives wrongly removed. In other words, for a fixed recall, I would be able to achieve higher precision values.

3.2 Results

Using different combinations of the developed approaches, five runs were submitted to the BioCreative IV CHEMDNER challenge. Each run combined different corpora and different validation processes. I used the CHEMDNER corpus and two external corpora for run 3, while only the CHEMDNER corpus was used for run 3*. These two runs provide the maximum recall achieved, since no validation process was employed. Run 3* was not submitted to the competition since the recall obtained with run 3 was higher, and there was a limit of five runs per team. Run 2 combines the CHEMDNER corpus and a high validation threshold based on the CRF confidence, ChEBI mapping score and semantic similarity to other entities in the same document. These three values were also used to train a Random Forest classifier to validate the CRF results, which corresponds to run 1. Run 4 uses only the CHEMDNER corpus, like run 3*, but each result is validated with semantic similarity, while run 5 uses the same training corpora as run 3, but

also with the semantic similarity validation. Each run is described with more detail in Section 3.1.

With the results from each run, I was able to generate predictions for the CEM subtask, using every entity recognized, and for the CDI subtask, considering only unique entities for each document. The metrics for each set of predictions were calculated using the official evaluation script on the results of 3-fold cross-validation for the CHEMDNER training and development dataset (Table 3.3). The official evaluation results are presented in Table 3.4. Generally, the results for the test set are better than using cross-validation.

Table 3.3: Precision (P), Recall (R) and F-measure (F) estimates for each approach used, using cross-validation on the CHEMDNER training set. The Approach column references the resources used, besides the CHEMDNER corpus, and the validation process applied, if any.

Run	Approach	CDI			CEM		
		P	R	F	P	R	F
1	DDI/PAT + RF	84.1%	72.6%	77.9%	87.3%	70.2%	77.8%
2	COMBINED	95.0%	6.5%	12.2%	95.0%	5.9%	11.1%
3	DDI/PAT	52.1%	80.4%	63.3%	57.1%	76.6 %	65.4%
3*	CHEMDNER only	76.7%	75.7%	76.2%	80.2%	72.8 %	76.3%
4	SSM	87.9%	22.7%	36.1%	89.7%	21.2%	34.3%
5	DDI/PAT + SSM	87.8%	22.7%	36.1%	79.9%	22.6%	35.3%

Table 3.4: Precision (P), Recall (R) and F-measure (F) estimates for each approach used, on the CHEMDNER test set. The Approach column references the resources used, besides the CHEMDNER corpus, and the validation process applied, if any.

Run	Approach	CDI			CEM		
		P	R	F	P	R	F
1	DDI/PAT + RF	85.3%	68.9%	76.2%	87.8%	65.2%	74.8%
2	COMBINED	96.8%	8.06%	14.9%	96.7%	7.11%	13.3%
3	DDI/PAT	57.7%	81.5%	67.5%	63.9%	77.9 %	70.2%
4	SSM	91.9%	24.4%	38.6%	92.9%	22.7%	36.4%
5	DDI/PAT + SSM	77.1%	27.3%	40.3%	79.7%	25.0%	38.1%

3.2.1 Best features

The precision, recall and F-measure values obtained using the four original features of ICE plus one new one are presented in Table 3.2.1. For each metric, a shaded column was added which compares that value with the one obtained on Table 3.4, for the run with best precision (run 2).

Table 3.5: Precision, Recall and F-measure estimates for each new features used with the original set, obtained with cross-validation on the CHEMDNER training set, for the CEM subtask

Feature set	P	ΔP	R	ΔR	F_1	ΔF_1
Prefix/suffix 1	92.4%	-2.6%	13.4%	+7.4%	23.4%	+12.3%
Prefix/suffix 2	93.5%	-1.5%	**18.3%**	+12.3%	**30.6%**	+19.5%
Prefix/suffix 4	94.2%	-0.8%	6.6%	+0.6%	12.2%	+1.1%
Greek letter	94.2%	-0.8%	11.8%	+5.8%	20.9%	+9.8%
Periodic table	94.7%	-0.3%	16.4%	+10.4%	28.0%	+16.9%
Amino acid	**95.1%**	+0.1%	8.7%	+2.7%	16.0%	+4.9%
Alphanumeric	92.0%	-3.0%	4.4%	-1.6%	8.4%	-2.7%
Case pattern	93.5%	-1.5%	14.9%	+8.9%	25.6%	+14.5%
Word shape	93.3%	-1.7%	12.7%	+6.7%	22.4%	+11.3%
Simple word shape	92.4%	-2.6%	16.9%	+10.9%	28.7%	+17.6%

The features that returned the best recall and F-measure were the simple word shape and prefix and suffix with size=2. Using prefix and suffix with size=1 and the alphanumeric boolean decreased the precision the most, without improving the other metrics as much as other features. The periodic table feature, which was one of the two domain-specific features, achieved a recall value of 16.4%, while maintaining the precision at 94%. The other domain-specific feature, amino acid, achieved the highest precision in this work. The general effect of using five features instead of the original four was a decrease in precision by 0.8%-4.5% and increase in recall and F-measure by 0.4%-19.5%.

I performed another cross-validation run with the original four features to use as baseline values. Based on these results, three feature sets were created, composed by the original features I used for BioCreative and the features that improved precision, recall or F-measure on any subtask, compared to the baseline. The three feature sets created were:

- **Best precision:** Stem, Prefix/suffix 3, Has number, Prefix/suffix 4, Has Greek symbol, Has periodic table element, Has amino acid.

- **Best recall:** Stem, Prefix/suffix 3, Has number, Prefix/suffix 1, Prefix/-suffix 2, Has Greek symbol, Has periodic table element, Case pattern, Word shape, Simple word shape.

- **Best F-measure:** Stem, Prefix/suffix 3, Has number, Prefix/suffix 1, Prefix/suffix 2, Has Greek symbol, Has periodic table element, Has amino acid, Case pattern, Word shape, Simple word shape.

The results obtained with these sets are presented in Table 3.2.1 Although there was a decrease in precision in every case, the difference in recall and F-measure values was always much higher. The feature set with best F-measure was able to improve the recall by 21.0% while taking only 3.2% of the precision. This feature set was then integrated in the module, and used for the following validation experiments.

Table 3.6: Precision, Recall and F-measure estimates for each feature set used with the original set, obtained with cross-validation on the CHEMDNER training set.

Feature set	P	ΔP	R	ΔR	F_1	ΔF_1
Precision	**94.1%**	-0.9%	15.0%	+9.0%	25.9%	+14.8%
Recall	92.0%	-3.0%	23.9%	+17.9%	37.9%	+26.8%
F-measure	92.3%	-2.7%	**28.0%**	+22.0%	**43.0%**	+31.9%
All features	93.0%	-2.0%	24.2%	+18.2%	38.4%	+27.3%

3.2.2 H-index for the ChEBI ontology

The h-index of each concept of the ChEBI ontology was computed. Figure 3.2 shows the average percentage of ancestors with an h-index above each threshold. We can see that about 10% of ancestors have an h-index higher than 7; based on this results, I decided to use the proposed measure with h-index of 2, 3, 4, 5 and 6. This decision was further validated when the results in Table 3.7 were obtained. In fact, once an h-index threshold of 6 is applied, precision values start to decrease, suggesting that the SSM scores start to degrade because of the high amount of concepts removed from the ancestry.

I tested each measure for different validation thresholds, obtaining different precision and recall values for each threshold and each SSM. As the validation

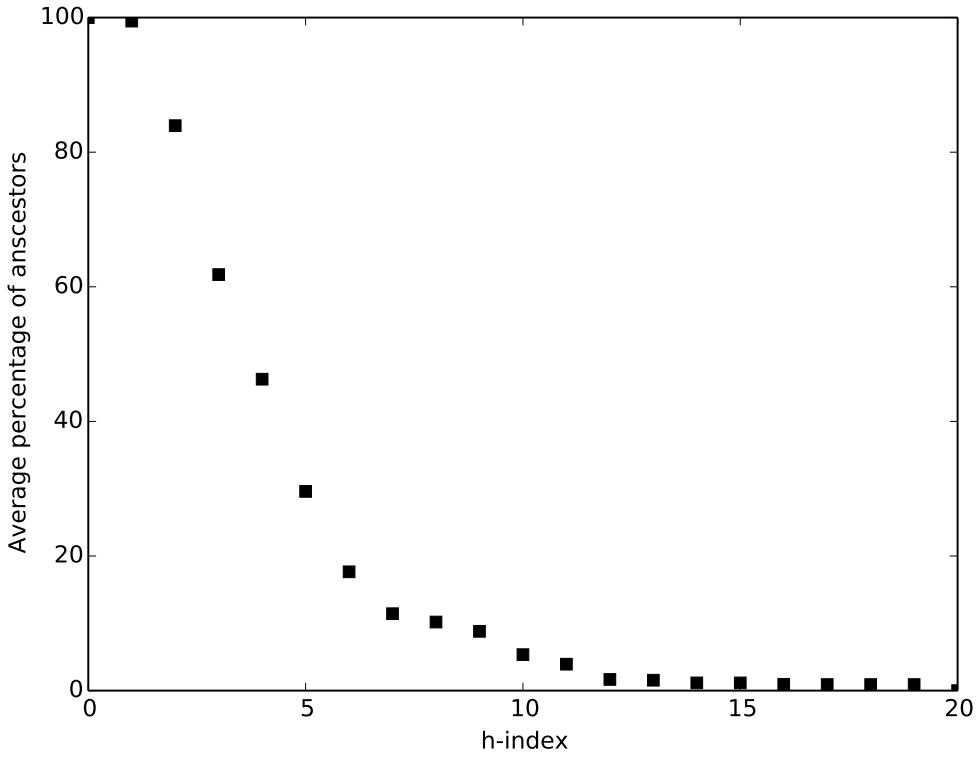

Figure 3.2: Average percentage of ancestors discarded using each h-index value.

threshold is increased, ideally the precision should also increase without affecting the recall. Eventually, true positives are also eliminated by this process, lowering the recall as the validation threshold increases. Figure 3.3 compares the precision and recall values obtained for different validation thresholds between simUI and simGIC and my proposed approach with five different h-index values. I restricted the recall values between 15% and 30%, since this is where the most of the points lie. Using my proposed approach, I obtained generally higher precision values for the same recall. This indicates that using the h-index information to measure semantic similarity results in a better performance at filtering out false positives from Machine Learning results. Furthermore, as the h-index increases, the difference between the original and the adapted measure increases. While on plot A of Figure 3.3, the points are mostly overlapping, this is less frequent on plot B, as the h-index measure achieves higher precision values. Between plots C,

D and E, this difference is less noticeable, which indicates that for higher h-index values, the filter becomes less efficient.

To confirm that the new adapted measures performed better at excluding fewer true positives, I compared the precision value obtained for each measure, with a fixed recall of 20%, on table 3.7. The points from Figure 3.3 that were closest to a recall of 20% were selected. Between each measure, the precision correspondent to similar recall values improves with the h-index used for the measure.

Table 3.7: Precision values obtained with each SSM for a fixed recall.

	P	R
simUI	92.97%	20.31%
$simUI_2$	93.14%	20.23%
$simUI_3$	93.01%	19.73%
$simUI_4$	93.10%	19.77%
$simUI_5$	93.35%	19.81%
$simUI_6$	93.00%	20.16%
simGIC	92.95%	20.23%
$simGIC_2$	93.14%	20.23%
$simGIC_3$	93.23%	19.85%
$simGIC_4$	93.24%	20.09%
$simGIC_5$	93.19%	20.10%
$simGIC_6$	93.10%	19.79%

3.2.3 Final evaluation

Table 3.8 shows the results obtained for the CHEMDNER and DDI gold standards, with the methods described in this section. I considered true positives only the entities that matched exactly the offsets of the gold standard, and did not attempt to classify the type of entity, which was required only for the DDI task. In this table, ICE 2013 refers to the best results obtained previously with that corpus, for the respective competition. For the DDI task, it corresponds to the results on Grego *et al.* (2013) while for the CHEMDNER task, it corresponds to the results of run 1 on Table 3.4. The classifiers used for each test set were the same, and were trained with both corpora. The validation processes in the rows

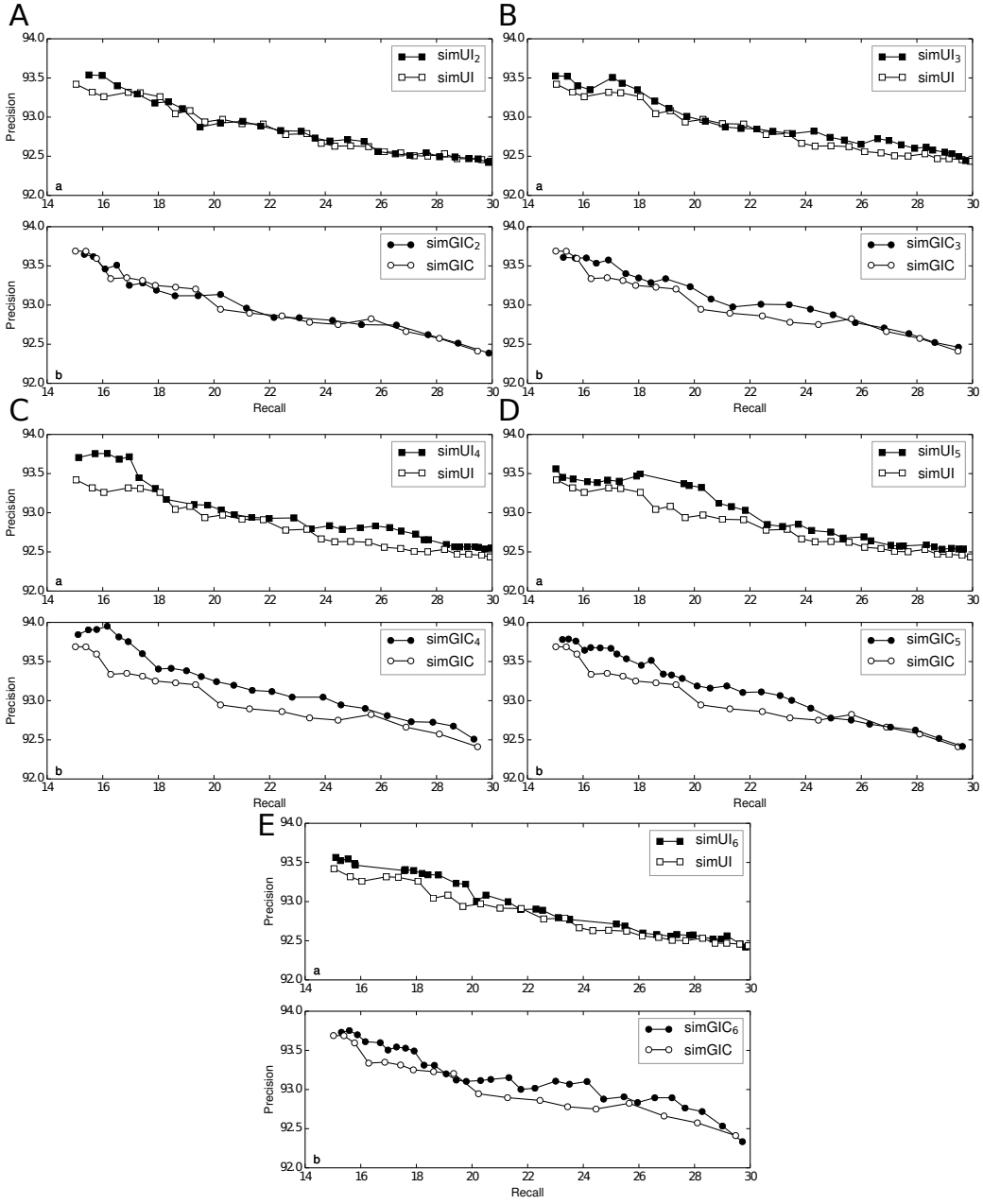

Figure 3.3: Comparison of precision and recall values for different thresholds between simUI and simGIC and variants with h-index $\geq$ 2,3,4,5 and 6, corresponding to the plots A, B, C, D and E, respectively.

of the table refer to the ones mentioned in the Methods section: "SSM" consists in filtering by one score, in this case, the SSM score; "COMBINED" consists in

filtering by a combination of scores, in this case, the average of the three highest scores for each results; and "RF" refers to the Random Forest classifier.

Table 3.8: Precision, Recall and F-measure estimates for NER using different validation processes, on the test set of the CHEMDNER and DDI corpus.

	CHEMDNER			DDI		
	P	R	F	P	R	F
ICE 2013	87.80%	65.20%	74.80%	82.80%	73.90%	78.10%
No validation	58.18%	**80.93%**	67.70%	79.40%	**81.49%**	80.43%
SSM	77.36%	46.64%	58.20%	86.56%	31.92%	46.65%
COMBINED	68.96%	33.13%	44.76%	**91.25%**	56.27%	69.61%
RF	**88.25%**	70.31%	**78.26%**	89.25%	76.24%	**82.23%**

On the CHEMDNER corpus, the best F-measure was of 78.26%, using the Random Forests validation, which is an improvement over the previous best F-measure (74.80%). The best F-measure on the DDI Corpus was of 82.23%, also with the Random Forests validation. On the DDI corpus, the results were higher than on the CHEMDNER corpus. However, without the improvements described on this chapter, the ICE framework also performed better on the DDI corpus.

3.3 Discussion

The results of runs 3 and 3* of Table 3.4 show the performance of the CNER module without any validation process. The values obtained are comparable with other applications of Mallet to this same task, for example, Campos *et al.* (2013). Since run 3 uses external corpora, the precision is much lower than run 3*, which uses only the CHEMDNER corpus. With each validation process, corresponding to the other four runs, I was able to improve precision, while run 1 also improved the F-measure of the CEM task by 4.6% on the test set. Every validation process also lowered significantly the recall, between 12%-60%. For this reason, I focused my work on improving the validation process so that the effect on recall is reduced.

Comparing with the results from other teams that participated on the CHEMDNER challenge, I achieved high precision values, especially on run 2 (96.8% for the CDI task), which was the second highest of all teams. However, the recall

obtained with that run was also one of the lowest of the competition. The results of this run should be viewed as an extreme case for the proposed validation process, since too many true positives were wrongly filtered out from the final result. Using semantic similarity (run 4), high precision were also achieved, without lowering the recall as much as run 2. The validation processes employed should be improved so that high precision values are still obtained, with minimal effect on the recall.

Individually, the implemented features that were specific to chemical compounds achieved the best balance between precision and recall. Adding only the prefixes and suffixes with size 2, I was able to increase the recall and F-measure by 12.3% and 19.5%, while decreasing the precision by 1.5%. Using a combination of the features that achieved the best results individually, I was able to increase the recall and F-measure by 21.2% and 31.0% respectively while decreasing the precision by 2.6% (Table 3.2.1).

By using the h-index to improve the simUI and simGIC measures, I was able to filter out fewer true positives with the validation process, and achieve higher precision values for the same recall. Comparing the simGIC with the simUI measure, which does not take into account the information content, the former measure achieved better results. The improvement is relatively small, but this may be because the NER applied was already well tuned for precision. This is an indication that the h-index provides a good estimate for the relevance of a class for the computation of the semantic similarity between two classes.

3.3.1 Error analysis

Analyzing the false positives committed by the CNER module on the CHEMD-NER corpus, it was possible to see that a common source of error were words that have prefixes and suffixes similar to chemical entities. For example, “nanoparticles”, “insulin”, “nanostructures” and “cytokines” were some of the most common false positives. Another source of errors were acronyms that do not refer to chemical entities, for example, “RNA”, “NMR” and “SAR”.

Regarding false negatives, even though one feature related to the periodic table was implemented, not all periodic table elements were identified, missing

49/80 mentions to "Ca(2+)", 26/99 mentions to "N" and 25/74 mentions to "C". This is due to the fact that these symbols are very ambiguous and it is necessary to understand completely the context of the token to distinguish between the chemical element and the letter.

The sources of false positives for the DDI corpus were similar to the ones described previously. However, it was possible to find some terms that were considered relevant on the CHEMDNER corpus but not on this one, for example, "warfarin", "ketoconazole", "fluconazole" and "lithium". This corpus was more focused on chemical entities with pharmacological interest, and with potential for interactions. However, the CHEMDNER classifiers also increased greatly the recall of the module on the DDI corpus, as it is possible to see on the first two lines of Table 3.8.

3.3.2 Limitations to other domains

The types of entities identified by this module are restricted to what was annotated on the corpora used to train the classifiers. However, the Mallet implementation of CRFs can be applied to any type of annotated and tokenized text.

The different validation processes employed depend on the ChEBI ontology, which is a domain-specific resource. In order to adapt to another domain, the recognized entities would have to be mapped to an appropriate database identifier. However, the same mapping process can be used for a different database or ontology, since it was originally developed for the Gene Ontology (Couto *et al.*, 2005).

The Random Forests classifier uses the classifier confidence score, mapping score and semantic similarity score as features. As long as these three scores are still provided for each putative entity, the results obtain with the RF validation process should be similar.

Chapter 4

Extraction of Chemical Interactions

Chemical interactions are described in scientific literature and can be a source of information for databases and ontologies. In this chapter I propose a module for the extraction of these chemical interactions. Since the previous chapter presented a module for the recognition of chemical entities mention in a given text, the input of this module is a biomedical document, annotated with chemical entities. The Chemical Interaction Extraction (CIE) module proposed here can be used by itself, or in conjunction with the CNER module.

4.1 Methods

Considering all the chemical entities annotated in a given text, each pair of entities mentioned in the same sentence is a potential interaction. Then, each pair of entities is classified as a true or false interaction and labeled with one of the DDI types considered in the DDI corpus. A Machine Learning classifier was trained to perform this classification, integrated with domain-specific resources. This module is able to bypass the CNER module and identify interactions in a text that is already annotated with chemical entities.

4.1.1 Pre-processing

As a pre-processing step, this module runs the input text through Stanford CoreNLP to extract additional information provided by this tool:

- Part-of-speech (POS) tagging;
- Parse tree;
- Co-reference resolution: the co-reference annotator is used to replace implicit references to a chemical entity by the representative words. This way, the structure of the sentence is simpler and easier to understand for a classifier. Co-reference resolution was considered to be one of the main source of errors in this task (Segura-Bedmar *et al.*, 2014);
- Named entity recognition: used to detect mentions to numbers, percentages and dates, which can improve the recall since drug interactions are often described with dosages and temporal references (Segura-Bedmar *et al.*, 2014).

Figure 4.1 provides an example of the pre-processing method and the type of data generated, which is then used as input for the Machine Learning classifiers.

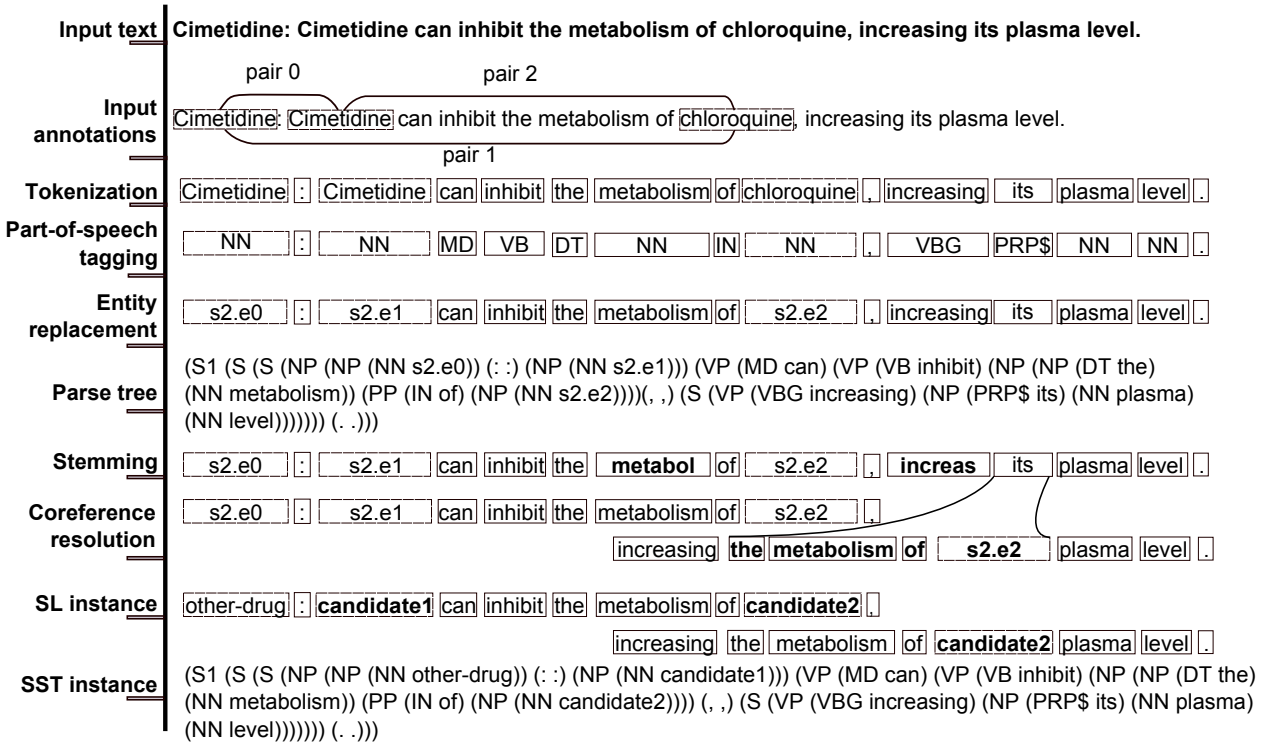

Figure 4.1: Pre-processing transformations on the input text for the CIE module.

The names of the chemical entities are replaced in the text by an identifier unique for each sentence. When classifying a pair, the identifiers of the two candidate entities are replaced by a generic string, and all the other chemical entities by a different generic string. This technique has been shown to improve the results of RE systems by ensuring the generality of the classifiers (Pyysalo *et al.*, 2008).

4.1.2 Machine Learning for pair classification

Kernel methods have gained popularity in the RE field and were employed by the teams that achieved the best results at the DDI Extraction task (Chowdhury & Lavelli, 2013b; Thomas *et al.*, 2013). A brief explanation of this type of methods is given on Section 2.1.2

I applied the Shallow Linguistic (SL) kernel, implemented by the jSRE tool (Giuliano *et al.*, 2006) and the SubSet Tree kernel (SST), implemented by the SVM-Light-TK toolkit (Joachims, 1999; Moschitti, 2006) to classify each pair instance.

The SL kernel is composite kernel that takes into account both the local and global context of the pair elements. I followed the recommendations provided by Segura-Bedmar *et al.* (2011), on which this kernel was also applied to the DDI corpus, obtaining good results which I intended to improve upon. Each training instance of this kernel is the whole sentence tokenized, where the two candidates are assigned a role of "Agent" and "Target". Whenever a candidate was mentioned more than once, by resolving co-references, an instance was added for each combination between the two pairs. This means that the example on Figure 4.1 would generate 5 instances: 3 for each pair and then 2 more where the second reference to the "s2.e2" entity is considered. Since the interactions considered were symmetric, "Agent" was always the first candidate and "Target" the second. This kernel calculates the similarity between two instances by comparing the text, POS tags, stems and label of each token. As such, I used the tokenization, POS tagging and stemming rows from Figure 4.1 besides the SL instance line, for the SL kernel. The label of each token was given by the Stanford NER, which could be only "NUMBER", "DATE", "PERCENTAGE", or "OTHER". For every chemical

entity, including the ones that did not constitute the pair, the label "DRUG" was assigned.

The SST kernel is a tree kernel that calculates the similarity between two instances by computing the number of common subset trees between two trees. For this kernel, the input is the smallest tree that contains both candidates (SST line of Figure 4.1) and the default parameters of the tool.

Both kernel methods classify each pair as interacting or not. One classifier was trained for each kernel method and for each type of interaction, as well as for the whole corpus, resulting in a total of 10 classifiers (4 types of interaction + 1 with the whole corpus for each of the two kernel methods).

4.1.3 Ensemble classifier

Even though the results of the kernel classifiers can directly classify the pairs, I implemented an ensemble SVM classifier, which uses as features the output of each RE classifier, along with a set of lexical and domain specific features. I used the SVM implementation of scikit-learn, based on LIBSVM, to train and test this classifier The feature set can be organized in three different groups: output of the kernel classifiers, ontological knowledge and presence of certain stems in the sentence. Table 4.1 shows a summary of the features used for this classifier.

Table 4.1: Feature set for the ensemble classifier, divided in three groups.

Kernel results		Ontological	Presence of stems in the sentence		
Kernel	DDI type				
SL	all	Resnik	advanc	advic	affect
	effect	simUI	anaesthetis	augment	awar
	mechanism	simGIC	bound	care	coadminist
	advice	$simUI_4$	combin	concentr	decreas
	int	$simGIC_4$	effect	exagger	expos
SST	all	ChEBI synonym	inhibit	ioniz	lengthen
	effect	ChEBI Distance	mechan	metabol	not
	mechanism	DrugBank interactions	note	part	prevent
	advice		reach	regul	short
	int		should	warn	withdrawn

The features derived from the classifiers could only be 0 or 1, depending on if the pair was classified as interacting or not. For example, if the SL kernel classifier trained with the type "effect" identified the pair as a true interaction, the feature "SL effect" would be equal to 1 for this instance. Since SSM values have been useful before for filtering false positives on the CNER module, this information is used again for the ensemble classifier in this module. I used five different SSMs as features: Resnik, simUI, simGIC, $simUI_4$ and $simGIC_4$, which I had already implemented for the CNER module. Moreover, three features based on DrugBank and ChEBI were added to improve the performance of the classifier:

- One candidate is a synonym of the other according to the ChEBI ontology
- Distance between the two candidates if one is an ascendant of the other in the ChEBI ontology (-1 otherwise)
- DrugBank entry for one candidate mentions the other candidate in the list of interactions

As some terms are more commonly employed than others when describing a type of interaction, I compiled a list of 32 stems that suggest the possibility of a DDIs, and added one binary feature for the presence of each word of this list. Finally, there is also another binary feature that has value 1 if the text of the two candidates is the same, since usually these pairs are not interactions.

This classifier was trained to label each pair with one of the following labels: "mechanism", "effect", "advice", "int" (the four DDI types considered in the training data) or "no-ddi", corresponding to pairs that do not represent an interaction. Finally, I used the evaluator released by the organization of the DDI Extraction task to compute the standard precision, recall and F-measure values.

4.2 Results

To evaluate the CIE module, I compared the results obtained with only the kernel methods, to the results obtained using also the ensemble classifier. In the first case, I considered a true DDI any pair classified as such by at least one classifier. If it was classified by more than one type-specific classifier, or only

by the whole-corpus classifier, I selected the type that was most frequent in the training data. The order of types, from most to least frequent, was: "effect", "mechanism", "advice" and "int". Otherwise, the DDI type was the one of the classifier that identified that DDI.

Two types of task were evaluated: the detection task consisted in simply labeling each pair as a DDI or not, while the classification task consisted in classifying each pair with one type of DDI or none. Table 4.2 shows the results obtained by training the classifiers with the training set and then testing on the test set. The ensemble classifier improved the precision of results for the detection and classification tasks, and also the F-measure of the classification task. The best F-measure for the detection task was 74.57%, using only the kernel methods, and for the classification task it was 64.02%, using the ensemble classifier.

Table 4.2: Precision, Recall and F-measure estimates for the CIE module, the test set of the DDI corpus.

	Task	P	R	F
Kernel	Detection	70.32%	79.37%	74.57%
	Classification	49.95%	56.38%	52.98%
Ensemble	Detection	80.20%	66.19%	72.52%
	Classification	70.79%	58.43%	64.02%

4.3 Discussion

My assumption was that an ensemble of classifiers and features would provide better results than using only one Machine Learning algorithm. In fact, just by using the two kernel methods, an acceptable F-measure was obtained, since the recall was maximized with this strategy. The kernel results provide a baseline for the ensemble classifier.

Without the ensemble classifier, higher recall values were achieved, since the positive pairs of two different classifiers were merged, but at the cost of lower precision. This classifier was able to generally increase the precision, particularly on the classification task. This task was more complex and, for this reason, the results were considerably lower: the highest F-measure for detection was 74.57%

while for classification, it was 64.02%. However, the ensemble classifier was able to reduce the difference between the F-measure of detection and classification by 12.11 percentage points on the train set and 13.48 percentage points on the test set. The main factor for this reduction was the increase in precision by the ensemble classifier, which uses Machine Learning to label the pairs with a DDI type. The ensemble classifier improved both precision and recall of the classification task. While it is still 7.76 percentage points lower than the recall of the detection task, this is an improvement over the classification results of the kernel methods.

However, the main advantage of the ensemble classifier was that it assigned the DDI types with more precision than the rule used for merging the results of the kernel methods. Hence, were able to increase the precision by 20.84 percentage points for the classification task. The results obtained were close to the best team of the detection and classification tasks of the DDI Extraction challenge (F-measure of 80.0% and 65.1%, respectively). Even though it did not achieve better results than the top systems of these competitions, this module is almost independent of external sources, using only the ChEBI ontology and DrugBank for domain knowledge.

4.3.1 Error analysis

Analyzing the false positives committed by the CIE module, I verified that many were caused by coordinate structures that were not resolved correctly by the parser. When one entity interacts with another, and then a list of examples for the second entity is provided, the module may not identify the interactions between the first entity and the list. For example, in the sentence "The induction dose requirements of DIPRIVAN Injectable Emulsion may be reduced in patients with intramuscular or intravenous premedication, particularly with narcotics (eg, morphine, meperidine, and fentanyl, etc.)", the module identified the interaction between "DIPRIVAN" and "narcotics", but not between "DIPRIVAN" and each of the narcotics mentioned.

Furthermore, the approach applied for resolving co-references is limited since it was not optimized for biomedical text, which can have complex sentence struc-

tures. In the sentence "It is reasonable to employ appropriate clinical monitoring when potent cytochrome P450 enzyme inducers, such as phenobarbital or rifampin, are co-administered with montelukast.", the module was unable to identify the interaction between "phenobarbital" and "montelukast".

I verified that 17 DDIs that the kernel methods were unable to identify were then correctly identified by the ensemble classifier using the domain and stem features. For example, the pair DDI-DrugBank.d585.s0.p2 of the DDI corpus, which is an interaction between "anticholinergic drugs" and "quinidine", was not identified by the kernel methods, possibly because of the complex structure of that sentence, which has 14 chemical entities, but the ensemble classifier correctly identified this pair as an interaction of the type effect.

4.3.2 Limitations to other domains

Even though this work was focused on the extraction of chemical interactions, the techniques used have been previously applied to other domains with success, such as protein-protein interactions and news articles. The first issue when applying to a different domain would be the corpora on which the kernel classifiers are trained. The natural language techniques employed are not specific to the biomedical domain, in fact, the models used by the Stanford CoreNLP toolkit are trained for the news domain. Nevertheless, domain-specific alternatives to the tools used to obtain the information from Figure 4.1 should provide even better results.

The kernel methods employed can be trained with any kind of corpus as long as it is annotated with the relevant entities. Although only one type of entities was considered on this work, it may be the case on other domains that different types of entities are mentioned in the text, and only some combinations of types may interact. This would require a pre-processing step to select the pairs that could be interacting according to this criteria. Then, these pairs would correspond to instances that can be used as input for the two kernel methods, as it was described in this chapter.

The ensemble classifier is the step most tuned for the chemical interactions domain. It employs domain-specific resources (ChEBI and DrugBank) as well as specific stems used to describe these types of interactions in scientific literature.

However, ontologies are available to other domains, and the semantic similarity measures used are not restricted to the ChEBI ontology and therefore can be applied to other ontologies. The list of stem expressions used to describe interactions would have to be adapted to a different domain. This list should take into account the different types of interactions considered in the domain.

Finally, an appropriate gold standard should be used to evaluate the performance on a different domain. This gold standard could be an independent partition of the corpus used for training, or a gold standard from a community challenge, for example.

Each domain has its own challenges which should be taken into account when adapting the methodology described in this chapter. The origin and number of features used by the ensemble classifier may have to be altered, and different kernel-based classifiers may also be added. However, this work provides a base framework for RE, achieving good results for the chemical interactions domain and it can possibly be adapted to other domains. For example, it may be applied to a large news corpus in order to extract interactions between persons, places, and organizations.

Chapter 5

IICE

5.1 Architecture

Combining the techniques developed and presented throughout Chapters 3 and 4, I developed a system for automatically Identifying Interactions between Chemical Entities (IICE) from biomedical text. An overview of the system architecture is presented in Figure 5.1. The system can process raw text without any annotation, or text already annotated with chemical entities, which is what I did to evaluate the RE module, starting the input on the box "Annotated Text" (step 4).

The first input of the system is one or more biomedical documents (1). These documents should contain information about chemical compounds and interactions, but it is not known where the chemical entities are located in the text. To analyze each document, it is first split by sentence, tokenize each sentence, and generate features for each token (step 2, Section 3.1.2). These features will be used by the CRF classifiers (step 3, Sections 2.5.1) to identify if each token or sequence of tokens refers to a chemical entity. Each chemical entity identified is then validated by one of the three processes described in Section 3.1.1 (step 4), which will employ external domain knowledge. At this point, each input document should be annotated with chemical entities. The steps 1-4 may be bypassed if the input documents are already annotated, manually or by a different system. As such, a pre-processing step is applied for extraction of chemical interactions (step 5, Section 4.1.1). Then, each pair of chemical entities is classified by the kernel classifiers (step 6, Section 4.1.2) as a true or false interaction. These results,

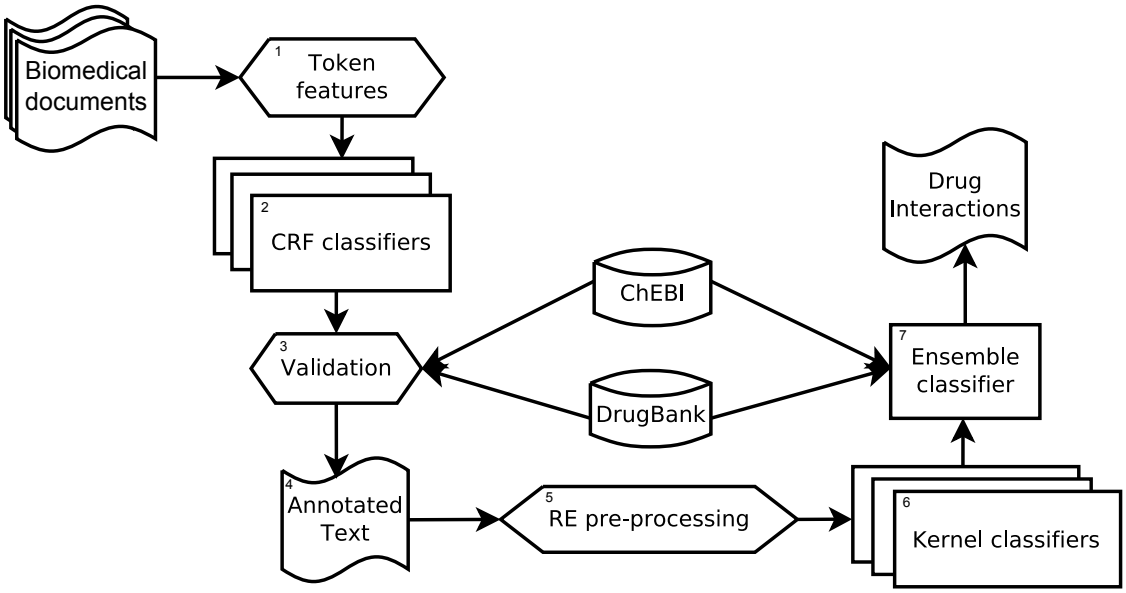

Figure 5.1: Overview of the system architecture.

along with external domain knowledge, are then used for the ensemble classifier (step 7, Section 4.1.3), which will assign a label to each pair, corresponding to a type of interaction, or none. The final result of this pipeline is the input documents annotated with chemical entities and interactions.

5.2 Implementation

The system was developed with Python programming language, version 2.6. At least one script was developed for each of the system components, represented as nodes on Figure 5.1. In same cases more than were script was developed, for example, one script was necessary for each kernel method, and another one to merge the results. Furthermore, two more scripts were developed to process the corpora for the challenges and train the classifiers. Finally, two scripts were developed to evaluate each module.

For the libraries used, preference was given to Python modules since these could be easily integrated with the main system. In same cases, there was no Python module, or it did not perform as well as another implementation of the same function. This was the case for the natural language processing tasks for

Relation Extraction, as well as the kernel classifiers. External libraries were integrated with system calls to Java classes, in this case, to Mallet, Stanford CoreNLP and Weka. The main Python libraries used were NLTK and ElementTree for simple text processing tasks, and sci-kit for general Machine Learning tasks. Each of the two corpora employed had one evaluation program, developed by the same authors, that I used to evaluate the results obtained.

These libraries implement complex Machine Learning algorithms or simple but useful tasks. However, the main challenge with the development of this system was the integration of these libraries with the input data. Each corpus was in a different format, and the expected format for the competitions was also different. To input to external libraries as to be written to a text file since they cannot be used directly with Python. Overall, the system processes the input data, performs pre-processing tasks both based on libraries and implemented anew, generates input for the Machine Learning libraries, reads the results and performs the final tasks necessary to generate the output.

For each task and corpus, the system can be called by command line to evaluate with cross-validation, train classifiers or test with new data. The input data can be provided in one of the formats adopted by the corpora, or as raw text in the command line. Three options are provided for the output format of the results: the HTML option is used to generate the tables for our web tool; the XML option corresponds to the same structure as the DDI corpus; the TSV option is similar to the format used for the CHEMDNER task, but adapted for interactions.

I implemented the command line options described on Table 5.1. The Steps column refers to the numbers on Figure 5.1 and show which steps of the pipeline are affected by each option.

The NER module has a series of options, related to the CRF classifiers and validation processes. It is possible to filter the predicted entities by Semantic Similarity Measure score (SSM), ChEBI mapping score (MAP), or by the confidence of the CRF classifier for that entity (CRF). The similarity measure can be chosen, from the ones described on Section 3.1.3. The best results were obtained with "simgic_hindex", which is my proposed version of the simGIC measure, considering only the most relevant ancestors. The COMBINED option is a filter on

Table 5.1: Description of the options available for the system.

Steps	Option	Values	Description
1,2,3	NER	Boolean	Recognize chemical entities mention in the text
5,6,7	DDI	Boolean	Identify drug-drug interactions in the text
2	Corpora	chemdner, ddi, all	Corpora to be used for entity recognition
3	Measure	resnik, simui, simgic, simui_hindex, simgic_hindex	Semantic similarity measure to be used for validation
		Validation	
3	SSM	Float	Thresold value for the SSM score
3	MAP	Float	Thresold value for the mapping score
3	CRF	Float	Thresold value for the CRF classifiers score
3	COMBINED	Float	Thresold value for the combined score
3	RF	Boolean	Use Random Forests classifier for entity recognition
		Relation Extraction	
6	Kernels	slk, sst	List of kernels to be used for DDI classification separated by commas
7	Ensemble	Boolean	Use ensemble classifier for entity recognition
4,8	Format	xml, html, tsv	Format for the results

a score which combines the SSM, MAP and CRF scores. However, Section 3.2.3 has shown that the best results are achieved with the Random Forests classifier, which corresponds to the RF option. This will use the previously trained Random Forests classifier to filter false positives. The classifiers were trained with two different corpora, CHEMDNER and DDI, each one being annotated with different guidelines. The input text can be classified with classifier from only one of these two, or both.

With the Relation Extraction module, it is possible to control the kernel classifiers used to classify the interactions and if the ensemble classifier is applied to the results. The two kernel methods described on Section 4.1.2 are available in the Kernels options. It is possible to use only one of them, although using both at the same time provide more robust results. The ensemble classifier will perform much better than the kernel classifiers at assigning the interaction types to the interactions identified, as shown on Table 4.2.

The system is more efficient when processing a large corpus than single documents or sentences. The test set for the CHEMDNER challenge consisted of 3000 abstracts and took approximately 24 hours to process, which results in an average of 29 seconds per abstract. However, it may take between one or two minutes to process a single abstract individually. One reason for this difference is the system calls to external libraries, which usually take more time that other instructions, and are only called once every time the system runs. This cost is less relevant if more documents are processed. The performance aspect of this system should be improved in the future, in order to be more efficient when processing individual documents.

5.3 Web tool

The web tool was developed in order to experiment the proposed system with a sentence or paragraph, available at `www.lasige.di.fc.ul.pt/webtools/iice`. Figure 5.2 shows screenshots of the input options, and results obtained with the web tool.

The user inserts a text to be analyzed in the text box. However, it is also possible to input text already annotated with chemical entities, by marking the

relevant entities with a "<entity>" tag. This is useful in case the user wants to determine how the Named Entity Recognition technique employed affects the identification of interactions. These tags will be considered only if the "NER" option is not checked. The tool can also analyze any PubMed abstract, just by inserting the PMID on the text box. In this case, the tool will automatically download the abstract from the PubMed webservices, and analyze that text.

The right panel on Figure 5.2 shows a series of options that can be changed. These options are related to Table 5.1 and serve two purposes: adjust the expected results in terms of types of entities (Corpora option) or precision (Validation) of the results; or experiment the influence of the details presented on Chapters 3 and 4, in terms of semantic similarity measures or kernel methods.

The output of this tool can be seen on the web page as one table with the chemical interactions and another table with the chemical entities, or it can be downloaded in XML format, similar to what was used for the DDI corpus. In case the user submits more than when sentence, the system will split by sentence and analyze each sentence individually. As such, one pair of tables is generated for each sentence, while the sentence to which they refer to is presented above them. The first table shows the interaction identified on the text: the two interacting chemical entities and the type of interaction. The second table shows all the chemical entities found, according to the thresholds established on Validation. In case the entity was mapped to ChEBI, a link is provided to the ChEBI page for that entity. A type of chemical entity is also provided, from the ones considered on the CHEMDNER corpus.

This web tool also provides additional information about the project. The "About" page is a brief explanation of the implementation of the system and resources used. The "Team" page shows describes the team that worked on this project, while "Publications" is a list the papers published about the methods described in this dissertation.

5.4 Conclusion

The work developed for this dissertation was combined on a chemical interaction extraction system for biomedical texts, entitled IICE. This system is able to

process text with and without annotations of chemical entities. A web tool was developed in order to access and test this system, with various options related to the techniques described. The main purpose of this tool is to demonstrate the capabilities of IICE. However, it may also be useful to curators performing semi-automatic annotations of biomedical documents. This system was recently presented on the Lisbon Machine Learning Summer School Demo Day 2014, on Instituto Superior Técnico.

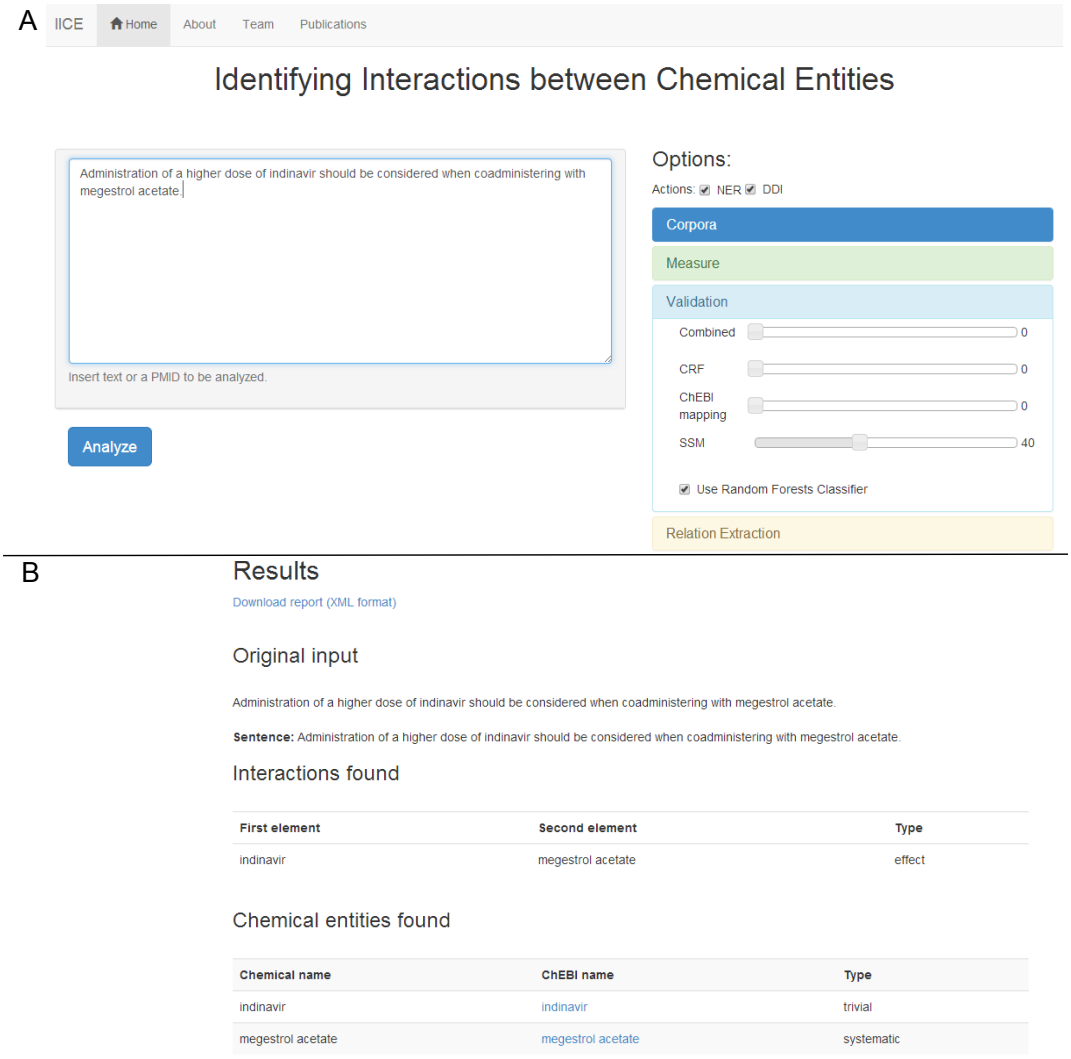

Figure 5.2: Screenshot of the Web tool. A: Input options; B. Results obtained.

Chapter 6

Conclusion

The hypothesis of this dissertation was to develop a system, based on two modules, to automatically and efficiently extract chemical interactions from biomedical text. I accomplished this by combining Machine Learning and Text Mining techniques with domain knowledge, to achieve higher levels of precision. The system was composed by two modules because first it is necessary to recognize the chemical compounds mentioned in a given text (CNER module), and only then the chemical interactions can be identified (CIE module).

The basis for the CNER module was an existing framework. I optimized the performance of this framework and evaluate on two corpora from community challenges. This module achieved a best F-measure of 82.23% and the maximum precision achieved was 91.25%, for a recall of 56.27%, on the DDI corpus. The F-measure value represent an improvement of 4.13 percentage points over the previous version of the framework, while being only 1.10 percentage points below the best performance of the competition. To improve the validation process, I developed a new category of semantic similarity measures based on the h-index, which filtered out fewer true positives, and achieved higher precision values for the same recall, compared to other measures.

The CNER module was based on two kernel methods applied to Support Vector Machines for Relation Extraction. The results obtained with the kernels were complemented with domain knowledge to train an ensemble classifier, in order to improve the classification of interactions. The best F-measure for detection of interactions was of 74.57%, obtained without the ensemble classifier. However,

this classifier obtained a F-measure of 65.02% on the classification of interactions task, which is an improvement of 11.04% percentage points over the F-measure obtained without the classifier, for the same task.

The whole IICE system was made available in the form of a web tool, which can be accessed at `www.lasige.di.fc.ul.pt/webtools/iice`. This web tool has a series of options which can be used to tune the results obtained for precision or recall. The input to this tool can be a PMID in order to analyze the abstract, or any other text that can be copied to the text box. This text can include entity annotation, in case the user wants to bypass CNER module. The results obtained with this tool can be downloaded in a XML file, with the same format as the DDI corpus. The IICE system has been shown to be effective at extracting information about chemical interaction from biomedical texts and was presented at the Lisbon Machine Learning Summer School Demo Day 2014, on Instituto Superior Técnico. This system opens the possibility of automatically analyzing old and new documents that are available, in order to construct or complement a database of chemical interactions, with minimal human intervention.

6.1 Future work

The work presented in this dissertation has been evaluated on two recent community challenges. However, the scope of these two challenges was limited, with focus on MEDLINE abstracts and DrugBank descriptions. The performance of the system may vary with different types of text, which is why it should be tested with other types of scientific literature. Furthermore, only one corpus for extraction of chemical interactions was used, which was focused on drug-drug interactions. Other types of chemical interactions should be explored in the future.

In order to overcome the lack of annotated corpus specific to this domain, unsupervised and semi-supervised Machine Learning algorithm should be explored. Deep Learning is one type of unsupervised learning algorithm that is currently being applied to Text Mining task mining tasks with success (Socher *et al.*, 2013b), and could possibly be integrated in the IICE system in the future.

The semantic similarity measure introduced could be further optimized by applying the h-index concept to other measures that fully explore the semantics

present in biomedical ontologies (Couto & Pinto, 2013). A better measure could improve the precision of the two developed modules since the assessment of the similarity between each pair of entities is crucial to both modules.

Even though this work was focused on chemical interactions, the techniques employed can and have been applied to other domains. In the future, I intend to adapt the whole system to the news domain. The idea is to extract relation between various types of entities, for example, persons, places and organizations. This system should perform better than pattern based approaches, obtaining results similar to what was obtained in this dissertation. The first step required to adapt the system for the news domain would be adapting the domain-specific techniques. For the news domain, the WordNet Fellbaum (1998) and DBpedia (Lehmann *et al.*, 2014) are two ontologies that should be explored. The semantic similarity measures mention in this dissertation would have to be implemented on those two ontologies.

References

Abarca, J., Malone, D.C., Armstrong, E.P., Grizzle, A.J., Hansten, P.D., Van Bergen, R.C. & Lipton, R.B. (2003). Concordance of severity ratings provided in four drug interaction compendia. *Journal of the American Pharmacists Association: JAPhA*, **44**, 136–141. 1

Aizerman, A., Braverman, E.M. & Rozoner, L. (1964). Theoretical foundations of the potential function method in pattern recognition learning. *Automation and remote control*, **25**, 821–837. 14

Apte, C., Damerau, F., Weiss, S. *et al.* (1998). *Text mining with decision rules and decision trees*. Citeseer. 13

Aronson, J. (2007). Communicating information about drug interactions. *British journal of clinical pharmacology*, **63**, 637–639. 1

Ashburner, M., Ball, C.A., Blake, J.A., Botstein, D., Butler, H., Cherry, J.M., Davis, A.P., Dolinski, K., Dwight, S.S., Eppig, J.T. *et al.* (2000). Gene ontology: tool for the unification of biology. *Nature genetics*, **25**, 25–29. 27

Banville, D.L. (2006). Mining chemical structural information from the drug literature. *Drug discovery today*, **11**, 35–42. 15

Batista-Navarro, R.T., Rak, R. & Ananiadou, S. (2013). Chemistry-specific features and heuristics for developing a CRF-based chemical named entity recogniser. In *BioCreative Challenge Evaluation Workshop vol. 2*, 55. 36

Bird, S., Klein, E. & Loper, E. (2009). *Natural language processing with Python*. " O'Reilly Media, Inc. ". 23

Björne, J., Heimonen, J., Ginter, F., Airola, A., Pahikkala, T. & Salakoski, T. (2011). Extracting contextualized complex biological events with rich graph-based feature sets. *Computational Intelligence*, **27**, 541–557. 29

Bundschus, M., Dejori, M., Stetter, M., Tresp, V. & Kriegel, H.P. (2008). Extraction of semantic biomedical relations from text using conditional random fields. *BMC bioinformatics*, **9**, 207. 12

Campos, D., Matos, S. & Oliveira, J.L. (2013). Chemical name recognition with harmonized feature-rich conditional random fields. In *BioCreative Challenge Evaluation Workshop vol. 2*, 82. 36, 47

Carletta, J. (1996). Assessing agreement on classification tasks: the kappa statistic. *Computational linguistics*, **22**, 249–254. 20

Chang, C.C. & Lin, C.J. (2011). LIBSVM: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, **2**, 27. 24

Charniak, E. (2000). A maximum-entropy-inspired parser. In *Proceedings of the 1st North American chapter of the Association for Computational Linguistics conference*, 132–139, Association for Computational Linguistics. 23

Chowdhury, M.F.M. & Lavelli, A. (2013a). Exploiting the scope of negations and heterogeneous features for relation extraction: A case study for drug-drug interaction extraction. In *HLT-NAACL*, 765–771, Citeseer. 29

Chowdhury, M.F.M. & Lavelli, A. (2013b). FBK-irst: A multi-phase kernel based approach for drug-drug interaction detection and classification that exploits linguistic information. *Atlanta, Georgia, USA*, 351. 53

Consortium, G.O. *et al.* (2012). The gene ontology: enhancements for 2011. *Nucleic acids research*, **40**, D559–D564. 27

Cortes, C. & Vapnik, V. (1995). Support-vector networks. *Machine learning*, **20**, 273–297. 3, 14

Coulet, A., Garten, Y., Dumontier, M., Altman, R.B., Musen, M.A., Shah, N.H. *et al.* (2011). Integration and publication of heterogeneous text-mined relationships on the semantic web. *J. Biomedical Semantics*, **2**, S10. 29

Couto, F.M. & Pinto, H.S. (2013). The next generation of similarity measures that fully explore the semantics in biomedical ontologies. *Journal of bioinformatics and computational biology*, **11**. 71

Couto, F.M., Silva, M.J. & Coutinho, P.M. (2005). Finding genomic ontology terms in text using evidence content. *BMC bioinformatics*, **6**, S21. 31, 49

Dickman, S. (2003). Tough mining. *PLoS biology*, **1**, e48. 10

Dietterich, T. (1995). Overfitting and undercomputing in machine learning. *ACM computing surveys (CSUR)*, **27**, 326–327. 14

DiGiacomo, R.A., Kremer, J.M. & Shah, D.M. (1989). Fish-oil dietary supplementation in patients with Raynaud's phenomenon: a double-blind, controlled, prospective study. *The American journal of medicine*, **86**, 158–164. 3

Fellbaum, C. (1998). *WordNet*. Wiley Online Library. 71

Finkel, J.R., Grenager, T. & Manning, C. (2005). Incorporating non-local information into information extraction systems by Gibbs sampling. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, 363–370, Association for Computational Linguistics. 23

Gentleman, R. (2005). Visualizing and distances using GO. *URL http://www. bioconductor. org/docs/vignettes. html*. 38

Giuliano, C., Lavelli, A. & Romano, L. (2006). Exploiting shallow linguistic information for relation extraction from biomedical literature. In *EACL*, vol. 18, 401–408, Citeseer. 24, 53

Greenhalgh, T. (1997). How to read a paper: The Medline database. *British Medical Journal*, **315**, 180. 1

Grego, T. & Couto, F.M. (2013). Enhancement of chemical entity identification in text using semantic similarity validation. *PloS one*, **8**, e62984. 6, 16, 29, 31, 36

Grego, T., Pęzik, P., Couto, F.M. & Rebholz-Schuhmann, D. (2009). Identification of chemical entities in patent documents. In *Distributed Computing, Artificial Intelligence, Bioinformatics, Soft Computing, and Ambient Assisted Living*, 942–949, Springer. 30

Grego, T., Pesquita, C., Bastos, H.P. & Couto, F.M. (2012). Chemical entity recognition and resolution to ChEBI. *International Scholarly Research Notices*, **2012**. 31

Grego, T., Pinto, F. & Couto, F.M. (2013). LASIGE: using conditional random fields and ChEBI ontology. *Proceedings of SemEval*, 660–666. 45

Gruber, T.R. (1993). A translation approach to portable ontology specifications. *Knowledge acquisition*, **5**, 199–220. 26

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P. & Witten, I.H. (2009). The WEKA data mining software: an update. *ACM SIGKDD explorations newsletter*, **11**, 10–18. 23

Hastings, J., de Matos, P., Dekker, A., Ennis, M., Harsha, B., Kale, N., Muthukrishnan, V., Owen, G., Turner, S., Williams, M. *et al.* (2013). The ChEBI reference database and ontology for biologically relevant chemistry: enhancements for 2013. *Nucleic acids research*, **41**, D456–D463. 27

Herrero-Zazo, M., Bedmar, I., Martínez, P. & Declerck, T. (2013). The DDI corpus: An annotated corpus with pharmacological substances and drug–drug interactions. *Journal of biomedical informatics*, **46**, 914–920. 5, 26

Hirsch, J.E. (2005). An index to quantify an individual's scientific research output. *Proceedings of the National academy of Sciences of the United States of America*, **102**, 16569. 39

Hirschman, L. & Blaschke, C. (2006). Evaluation of text mining in biology. In S. Ananiadou & J. McNaught, eds., *Text mining for biology and biomedicine*, 213–245, Artech House London. 18

Huber, T., Rocktäschel, T., Weidlich, M., Thomas, P. & Leser, U. (2013). Extended feature set for chemical named entity recognition and indexing. In *BioCreative Challenge Evaluation Workshop vol. 2*, 88. 36

Joachims, T. (1998). *Text categorization with support vector machines: Learning with many relevant features*. Springer. 14

Joachims, T. (1999). Making large scale SVM learning practical. 24, 53

Klein, D. & Manning, C.D. (2003). Accurate unlexicalized parsing. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*, 423–430, Association for Computational Linguistics. 23

Kohavi, R. *et al.* (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. In *IJCAI*, vol. 14, 1137–1145. 15

Krallinger, M., Leitner, F., Rodriguez-Penagos, C., Valencia, A. *et al.* (2008). Overview of the protein-protein interaction annotation extraction task of BioCreative II. *Genome biology*, **9**, S4. 12

Krallinger, M., Leitner, F., Rabal, O., Vazquez, M., Oyarzabal, J. & Valencia, A. (2014a). The CHEMDNER corpus of chemicals and drugs and its annotation principles. 24

Krallinger, M., Leitner, F., Rabal, O., Vazquez, M., Oyarzabal, J. & Valencia, A. (2014b). CHEMDNER: The drugs and chemical names extraction challenge. 21

Lafferty, J., McCallum, A. & Pereira, F.C. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. 4, 13

Lamurias, A. & Couto, F. (2014). Identifying interactions between chemical entities in text. In *Bioinformatics Open Days, University of Braga*. 6

Lamurias, A., Grego, T. & Couto, F.M. (2013). Chemical compound and drug name recognition using CRFs and semantic similarity based on ChEBI. In *BioCreative Challenge Evaluation Workshop*, vol. 2, 75. 6

Lamurias, A., Ferreira, J. & Couto, F.M. (2014a). Chemical named entity recognition: Improving recall using a comprehensive list of lexical features. In *8th International Conference on Practical Applications of Computational Biology & Bioinformatics (PACBB 2014)*, 253–260, Springer. 6

Lamurias, A., Ferreira, J. & Couto, F.M. (2014b). Identifying interactions between chemical entities in biomedical text. *Journal of Integrative Bioinformatics*. 6

Lamurias, A., Ferreira, J. & Couto, F.M. (2014c). Improving chemical entity recognition through h-index based semantic similarity. *Journal of cheminformatics*. 6

Law, V., Knox, C., Djoumbou, Y., Jewison, T., Guo, A.C., Liu, Y., Maciejewski, A., Arndt, D., Wilson, M., Neveu, V. *et al.* (2014). DrugBank 4.0: shedding new light on drug metabolism. *Nucleic acids research*, **42**, D1091–D1097. 27

Leaman, R., Gonzalez, G. *et al.* (2008). BANNER: an executable survey of advances in biomedical named entity recognition. In *Pacific Symposium on Biocomputing*, vol. 13, 652–663. 16

Leaman, R., Wei, C.H. & Lu, Z. (2013). NCBI at the bioCreative IV CHEMDNER task: Recognizing chemical names in PubMed articles with tmChem. In *BioCreative Challenge Evaluation Workshop vol. 2*, 34. 36

Lee, H., Chang, A., Peirsman, Y., Chambers, N., Surdeanu, M. & Jurafsky, D. (2013). Deterministic coreference resolution based on entity-centric, precision-ranked rules. *Computational Linguistics*, **39**, 885–916. 18, 23

Lehmann, J., Isele, R., Jakob, M., Jentzsch, A., Kontokostas, D., Mendes, P.N., Hellmann, S., Morsey, M., van Kleef, P., Auer, S. & Bizer, C. (2014). DBpedia - a large-scale, multilingual knowledge base extracted from Wikipedia. *Semantic Web Journal*. 71

Liu, H., Christiansen, T., Baumgartner Jr, W.A. & Verspoor, K. (2012). BioLemmatizer: a lemmatization tool for morphological processing of biomedical text. *J. Biomedical Semantics*, **3**, 3. 17

Manning, C.D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S.J. & McClosky, D. (2014). The Stanford CoreNLP natural language processing toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, 55–60. 23

McCallum, A.K. (2002). MALLET: A machine learning for language toolkit, http://mallet.cs.umass.edu. 24

McClosky, D. & Adviser-Charniak, E. (2010). Any domain parsing: automatic domain adaptation for natural language parsing. 23

Moschitti, A. (2006). Making tree kernels practical for natural language learning. In *EACL*, 113–120. 24, 53

CRFsuite: a fast implementation of conditional random fields (CRFs). 24

Paice, C.D. (1996). Method for evaluation of stemming algorithms based on error counting. *Journal of the American Society for Information Science*, **47**, 632–649. 17

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M. & Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, **12**, 2825–2830. 23

Pesquita, C., Faria, D., Bastos, H., Falcão, A. & Couto, F. (2007). Evaluating GO-based semantic similarity measures. In *Proc. 10th Annual Bio-Ontologies Meeting*, 37–40. 38

Porter, M.F. (1980). An algorithm for suffix stripping. *Program: electronic library and information systems*, **14**, 130–137. 17

Pyysalo, S., Sætre, R., Tsujii, J. & Salakoski, T. (2008). Why biomedical relation extraction results are incomparable and what to do about it. In *Proceedings of the Third International Symposium on Semantic Mining in Biomedicine (SMBM 2008). Turku*, 149–152. 53

Rennie, J.D., Shih, L., Teevan, J., Karger, D.R. *et al.* (2003). Tackling the poor assumptions of Naive Bayes text classifiers. In *ICML*, vol. 3, 616–623, Washington DC). 13

Seco, N., Veale, T. & Hayes, J. (2004). An intrinsic information content metric for semantic similarity in WordNet. In *ECAI*, vol. 16, 1089, Citeseer. 38

Segura-Bedmar, I., Crespo, M., de Pablo-Sánchez, C. & Martínez, P. (2010). Resolving anaphoras for the extraction of drug-drug interactions in pharmacological documents. *BMC bioinformatics*, **11**, S1. 18

Segura-Bedmar, I., Martinez, P. & de Pablo-Sánchez, C. (2011). Using a shallow linguistic kernel for drug–drug interaction extraction. *Journal of biomedical informatics*, **44**, 789–804. 24, 53

Segura-Bedmar, I., Martinez, P. & Herrero-Zazo, M. (2013). SemEval-2013 task 9: Extraction of drug-drug interactions from biomedical texts (DDIExtraction 2013). *Proceedings of Semeval*, 341–350. 5, 21, 28

Segura-Bedmar, I., Martínez, P. & Herrero-Zazo, M. (2014). Lessons learnt from the DDIExtraction-2013 shared task. *Journal of biomedical informatics*. 23, 52

Socher, R., Bauer, J., Manning, C.D. & Ng, A.Y. (2013a). Parsing with compositional vector grammars. In *In Proceedings of the ACL conference*, Citeseer. 17

Socher, R., Perelygin, A., Wu, J.Y., Chuang, J., Manning, C.D., Ng, A.Y. & Potts, C. (2013b). Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 1631–1642, Citeseer. 70

Swanson, D.R. (1990). Medical literature as a potential source of new knowledge. *Bulletin of the Medical Library Association*, **78**, 29. 3

Tan, A.H. *et al.* (1999). Text mining: The state of the art and the challenges. In *Proceedings of the PAKDD 1999 Workshop on Knowledge Disocovery from Advanced Databases*, vol. 8, 65–70. 10

Thomas, P., Neves, M., Rocktäschel, T. & Leser, U. (2013). WBI-DDI: Drug-drug interaction extraction using majority voting. In *Proceedings of the seventh international workshop on semantic evaluation (SemEval 2013)*, 628–635, Citeseer. 29, 53

Toutanova, K. & Manning, C.D. (2000). Enriching the knowledge sources used in a maximum entropy part-of-speech tagger. In *Proceedings of the 2000 Joint SIGDAT conference on Empirical methods in natural language processing and very large corpora: held in conjunction with the 38th Annual Meeting of the Association for Computational Linguistics-Volume 13*, 63–70, Association for Computational Linguistics. 23

Toutanova, K., Klein, D., Manning, C.D. & Singer, Y. (2003). Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, 173–180, Association for Computational Linguistics. 17

Tsuruoka, Y., Tateishi, Y., Kim, J.D., Ohta, T., McNaught, J., Ananiadou, S. & Tsujii, J. (2005). Developing a robust part-of-speech tagger for biomedical text. In *Advances in informatics*, 382–392, Springer. 17

Usié, A., Cruz, J., Comas, J., Solsona, F. & Alves, R. (2013). A tool for the identification of chemical entities (CheNER-BioC). In *BioCreative Challenge Evaluation Workshop vol. 2*, 66. 36

Webster, J.J. & Kit, C. (1992). Tokenization as the initial phase in NLP. In *Proceedings of the 14th conference on Computational linguistics-Volume 4*, 1106–1110, Association for Computational Linguistics. 16

Witten, I.H. & Frank, E. (2005). *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann. 12

Wong, P.C., Whitney, P. & Thomas, J. (1999). Visualizing association rules for text mining. In *Information Visualization, 1999.(Info Vis' 99) Proceedings. 1999 IEEE Symposium on*, 120–123, IEEE. 13

Zelenko, D., Aone, C. & Richardella, A. (2003). Kernel methods for relation extraction. *The Journal of Machine Learning Research*, **3**, 1083–1106. 13, 14