UNIVERSIDADE DE LISBOA
FACULDADE DE CIÊNCIAS
DEPARTAMENTO DE INFORMÁTICA

Automatic Extraction of Definitions

Rosa Del Gaudio

DOUTORAMENTO EM INFORMÁTICA
ESPECIALIDADE ENGENHARIA INFORMÁTICA

2013

UNIVERSIDADE DE LISBOA
FACULDADE DE CIÊNCIAS
DEPARTAMENTO DE INFORMÁTICA

Automatic Extraction of Definitions

Rosa Del Gaudio

Tese orientada pelo Prof. Doutor António Manuel Horta Branco
e pelo Prof. Doutor Gustavo Enrique de Almeida Prado Alves Batista, especialmente elaborada para a obtenção do grau de doutor em Informática, especialidade Engenharia Informática.

2013

Abstract

This doctoral research work provides a set of methods and heuristics for building a definition extractor or for fine-tuning an existing one. In order to develop and test the architecture, a generic definitions extractor for the Portuguese language is built. Furthermore, the methods were tested in the construction of an extractor for two languages different from Portuguese, which are English and, less extensively, Dutch. The approach presented in this work makes the proposed extractor completely different in nature in comparison to the other works in the field. It is a matter of fact that most systems that automatically extract definitions have been constructed taking into account a specific corpus on a specific topic, and are based on the manual construction of a set of rules or patterns capable of identifyinf a definition in a text.

This research focused on three types of definitions, characterized by the connector between the defined term and its description. The strategy adopted can be seen as a "divide and conquer"approach. Differently from the other works representing the state of the art, specific heuristics were developed in order to deal with different types of definitions, namely copula, verbal and punctuation definitions.

We used different methodology for each type of definition, namely we propose to use rule-based methods to extract punctuation definitions, machine learning with sampling algorithms for copula definitions, and machine learning with a method to increase the number of positive examples for verbal definitions. This architecture is justified by the increasing linguistic complexity that characterizes the different types of definitions. Numerous experiments have led to the conclusion that the punctuation definitions are easily described using a set of rules. These rules can be easily adapted to the relevant context and translated into other languages. However, in order to deal with the other two definitions types, the exclusive use of rules is not

enough to get good performance and it asks for more advanced methods, in particular a machine learning based approach.

Unlike other similar systems, which were built having in mind a specific corpus or a specific domain, the one reported here is meant to obtain good results regardless the domain or context. All the decisions made in the construction of the definition extractor take into consideration this central objective.

Keywords: Natural Language Processing, Information Extraction, Machine Learning, Rule-Based Methods, Definitions Extraction

Resumo

Este trabalho de doutoramento visa proporcionar um conjunto de métodos e heurísticas para a construção de um extractor de definição ou para melhorar o desempenho de um sistema já existente, quando usado com um corpus específico. A fim de desenvolver e testar a arquitectura, um extractor de definições genérico para a língua Portuguesa foi construído. Além disso, os métodos foram testados na construção de um extractor para um idioma diferente do Português, nomeadamente Inglês, algumas heurísticas também foram testadas com uma terceira língua, ou seja o Holandês. A abordagem apresentada neste trabalho torna o extractor proposto neste trabalho completamente diferente em comparação com os outros trabalhos na área. É um fato que a maioria dos sistemas de extracção automática de definições foram construídos tendo em conta um corpus específico com um tema bem determinado e são baseados na construção manual de um conjunto de regras ou padrões capazes de identificar uma definição num texto dum domínio especifico.

Esta pesquisa centrou-se em três tipos de definições, caracterizadas pela ligação entre o termo definido e a sua descrição. A estratégia adoptada pode ser vista como uma abordagem "dividir para conquistar". Diferentemente de outras pesquisa nesta área, foram desenvolvidas heurísticas específicas a fim de lidar com as diferentes tipologias de definições, ou seja, cópula, verbais e definições de pontuação.

No presente trabalho propõe-se utilizar uma metodologia diferente para cada tipo de definição, ou seja, propomos a utilização de métodos baseados em regras para extrair as definições de pontuação, aprendizagem automática, com algoritmos de amostragem para definições cópula e aprendizagem automática com um método para aumentar automaticamente o número de exemplos positivos para a definição verbal. Esta arquitetura é justificada pela complexidade linguística crescente que caracteriza os diferentes tipos de

definições. Numerosas experiências levaram à conclusão de que as definições de pontuação são facilmente descritas utilizando um conjunto de regras. Essas regras podem ser facilmente adaptadas ao contexto relevante e traduzido para outras línguas. No entanto, a fim de lidar com os outros dois tipos de definições, o uso exclusivo de regras não é suficiente para obter um bom desempenho e é preciso usar métodos mais avançados, em particular aqueles baseados em aprendizado de máquina.

Ao contrário de outros sistemas semelhantes, que foram construídos tendo em mente um corpus ou um domínio específico, o sistema aqui apresentado foi desenvolvido de maneira a obter bons resultados, independentemente do domínio ou da língua. Todas as decisões tomadas na construção do extractor de definição tiveram em consideração este objectivo central.

Palavras Chave: Processamento de Linguagem Natural, Extração de Informação, Aprendizagem Automática, Métodos Baseados em Regras, Extração de Definições

Acknowledgements

I wish to thank various people for supporting me during this project. All the colleagues at the NLX group were a valuable presence, their advices and their availability have helped me in the most difficult moments.

I would like to express my deep gratitude to Professor António Branco and Professor Gustavo Batista, my research supervisors, for their patient guidance, encouragement and useful critiques of this research work.

A very special thank goes to my partner for his support and encouragement and the delicious meals.

This research work work was funded by Fundação para a Ciência e Tecnologia, under the scholarship SFRH/ BD/36732/2007. This dissertation would not have been possible without this scholarship.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Defining a concept making use of expressions other than the ones conveying that concept itself is recognized to be one of the most valuable functions of language (Barnbrook, 2002). For this reason, probably, the interest in this issue dates back to the Antiquity, starting with Ancient Philosophy, when by the time of Plato and Aristotle definitions started to be discussed. The notion of definition has been investigated and used more than any other part of the original theory of Aristotle logic. Plato first and then Aristotle devoted a lot of effort in clarifying the notion of definition. In one of his dialogues, Plato (ca. 360 BC) depicted Socrates saying that "if you get at the difference and distinguishing characteristic of each thing, then, as many persons affirm, you will get a definition or explanation of it."[1] This is one of the passages where the most famous, discussed and then obviously criticized notion of definition is provided, that is of a definition as a statement composed by *genus* and *differetiae*.

Nowadays, definitions are central in almost all different knowledge fields, such as Logic, Mathematics, Law, etc. As it is such a relevant and powerful device, definitions have been used with so many different meanings that is fairly difficult to state what a definition is without specifying the context where it is used.

In the twentieth century, due to the progressive development of science, experts expressed their own particular interest in systematizing rules for terms in each domain of expertise. Both science and technology experienced an unprecedented rate of development, resulting in the emergence of new concepts and conceptual fields, which need to be regulated and standardized. The diffusion of new technologies within society

[1] Theaetetus, 208.

created new fields of action. This dissemination of technology is both cause and consequence of an unprecedented development in the field of communication and information dissemination, both monolingual and multilingual.

Moreover, the increasing interdisciplinarity of different sciences forces to eliminate possible ambiguities in the communication and led to the uniformization of the designations given, and the terms used to refer to concepts that belong to various disciplines. The development of mass media allows widespread dissemination of terminology, with the resulting interaction between the general and the specialized vocabularies. The dissemination of specialized information among non-specialists (such as, for instance, medical information) is increasingly common, so the need to clarify and explain the meaning of terms used by specialists is evident. This increase of information demanded the development of systematic and organized methods supporting clarification and standardization of concepts to facilitate communication.

The automatic extraction of definitions from specialized texts represents an answer to this challenging information and communication environment and to this need. Definitions extracted in this way are becoming an important source of knowledge and an invaluable support in the construction of dictionaries, thesauri, ontologies and lexica.

1.1 Definitions: a First Operational Definition

One of the most discussed aspects in the studies on definition is the concept of definition itself, thus addressing what should be considered a definition and what should not. In this Section, a brief operational description is given while a more detailed discussion is offered in Chapter 2.

In a definition, the object to be clarified (either a term or a symbol) is called *definiendum*, whereas the expression providing the clarification or the explanation is the *definiens*. In the sentence *Lightning conductor is an electronic device that allows to protect the electrical systems against surges of atmospheric origin*, *Lightning conductor* is the *definiendum* while the expression *an electronic device that allows to protect the electrical systems against surges of atmospheric origin* is the *definiens*. One can say that a definition is an assertion or a proposition where the *definiendum* conveys the same meaning as the *definiens*. Furthermore, the *definiens* may consist of two parts: the *genus*, conveying the nearest superior concept of which the concept being defined is

an instance or a subclass, and the *differentiae specificae*, the distinguishing characteristics that turn the denotation of the *definiendum* distinguishable from other instances or subclasses of the denotation of the *genus*. This characterization comes from Aristotle and is called analytical definition. In practical terms, analytical definitions are a very common mechanism for defining concepts and remain as a means of capturing and transferring knowledge in specialized domains.

Starting from this characterization, it is possible to reach several subtypes of definitions. For instance, Sierra *et al.* (2006) enumerate four types: exclusive genus, synonymic, functional and extensional. In the first type of definitions, only the genus is given, with no information about the distinguishing characteristics. Using the previous example, the definition would become *Lightning conductor is an electronic device*. Regarding the synonymic definition, in this case only a synonym is given; for instance, *Lightning conductor means the same that lightning rod*. In functional definitions, the *differentia* describe how to use or the purpose of what is described by the *definiendum*, as in the original example. The extensional definition type aims to describe the parts composing it or to enumerate the sub-concepts. For example, one can describe the lightning conductor as *a device composed by a long thin piece of metal on top of a building that attracts lightning*. Finally, definitions can be provided without providing the *genus*; in this case they are called semi-formal definitions (in opposition to formal definitions, where all the components are presented).

A more detailed analysis of definitions in the scope of an expert domain may take into consideration the fact that not all information is relevant to define a given concept. The focus is on how to determine the relevance of a given piece of information and in which usage context (Seppälä, 2009).

In the present work, a definition (also called definitory context) is assumed to be a sentence containing an expression, the *definiendum*, another expression, the *definiens* and a possible connector between them.

We will restrict our attention to three different and most common connectors: punctuation mark, such as the colon ":" and the dash "-", the verb *to be*, all other verbs other than *to be*. We will be calling punctuation definitions the ones introduced by punctuation marks, copula definitions all those definitions where the verb *to be* acts as a connector, and verbal definitions all those definitions that are introduced by a verb other than *to be*, as in the following examples:

- **punctuation definition**
  TCP/IP: protocols used in the transfer of information between computers.
- **copula definition**
  FTP is a protocol that allows the transfer of archives from a place to another through the Internet.
- **verb definition**
  An ontology can be described as a formal definition of objects.

Sentences not presenting this structure are not taken into consideration even if definitional information may be displayed, as in the following examples:

Software, the name given to the coded instructions that tell computers what to do, comes in many different forms .

This leads to what are called electronic spreadsheets (a tool of visualization and simulation used by bookkeepers, economists, and others).

1.2 Why Automatic Definition Extraction

Automatic definition extraction represents an important tool for supporting different applications such as dictionary and glossary building, question answering, knowledge management and ontology engineering. In the last decade, several systems were developed for these different purposes, as briefly introduced below (a detailed discussion will be undertaken in Chapter 3).

Dictionaries and glossaries

A milestone for any natural language is the creation of dictionaries for it. The importance of a dictionary lies in its function, allowing to look up the meaning of terms in a quick and effective way. One can expand his knowledge and understanding of a topic by relating the new terms encountered to the knowledge previously available.

Both general purpose dictionaries or domain specific glossaries can be considered as indispensable references. When one reads a book or an article, in particular in an area that he is not familiar with, one may find a word whose meaning is not clear, or that is used in a different sense in that particular context. A general purpose dictionary may not contain that word or may not list that specific meaning. Technical meanings are often neglected in dictionary entries and users often run into difficulties.

For this purpose, one needs a technical dictionary or a glossary providing definitions for those more technical terms occurring in the book or article. The construction of a glossary for a specific knowledge domain is thus useful, for example, (i) to systematize a knowledge domain, (ii) to facilitate learning and retention of concepts, and (iii) as a reference resource for searching for information in specific areas. Glossaries have been shown to facilitate reception of texts and acquisition of knowledge during study (Weiten *et al.*, 1999), while explanation by referring to definitions has been shown to promote understanding (Aleven *et al.*, 1999).

However, many texts or books rarely come with an accompanying glossary, since it requires substantial effort to be produced, with extensive work carried out by experts in the target field, though most of the definitions would be present in the text itself. The problem is even more acute in the context of e-learning. Normally in an e-learning environment, tutors make available to their students different learning materials, such as manuals, articles, or tutorials. Generally, these materials are not paired with a glossary, where terms are explained. This is because creating such glossaries is a very time consuming task, and usually tutors do not have the time to do it. In this learning scenario, students have to try to figure out the meaning of the concept, or they have to try to look for it in a dictionary, hoping that the right sense is listed, and they are able to select it. For these reasons, a tool to automatically locate definitions in texts is of great value. It allows tutors to compile a list of definitions on the basis of the texts they make available. This assures students to have access directly to correct meaning of the term and enables them to spend more time with the actual learning process.

Furthermore, glossaries are useful not only in a learning scenario but also in a translation scenario, as reported by Durán-Muñoz (2010), who present a survey on the use of lexicographical aids in specialized translation showing that glossaries are among the top five resources used.

A glossary is also a reference resource so that everyone has the same understanding of a term, allowing the reduction of communication failures and the building of a common language. Irrespective of the domain knowledge, the use of terminologies, jargons and specialized vocabularies arise naturally in response to the need for precise communication of concepts that are specific to target areas. This communication becomes more difficult with the constant technological and scientific evolution, resulting in a constant process where new terms arise, and old terms are dismissed or may undergo a shift in

their meaning. A way to deal with this evolution is to gather the terms from domain specific documents, along with their definitions. Accordingly, automatic extraction of definitions comes to assist in this task by processing the text and automatically finding terms and their definitions that could be used as-is or revised for the construction of glossaries.

Question Answering

A question answering (QA) system is an information retrieval application whose aim is to provide users with a more easy access to information, allowing them to write a query in natural language and obtaining not a set of documents that contain the answer, but the concise answer itself. In the scenario of a question answering system, one can automatically extract information on the basis of a particular type of question, called definition question, such as "What is X?".

QA systems, handling definition questions, are not only interesting as a research challenge, they also have the potential to be a valuable complement to static knowledge sources like encyclopedias. This is because they can create definitions dynamically, and thus answer definitional questions about new or emerging terms. They can also tailor an answer to a user's needs, for instance by creating a longer or shorter definition, or one which is drawn from specified knowledge sources (Blair-Goldensohn *et al.*, 2004).

The advent of the web has reintroduced the need for user-friendly querying techniques that reduce information overflow, and poses new challenges to the research in automated QA. Important QA application areas are information extraction from the entire Web (intelligent search engines), online databases, and inquiries on individual websites. The utility of such a tool is the extraction of knowledge, and possibly the ability of handling that knowledge for further use, such as in ontology engineering where terms and their meanings are used for automatic ontology creation.

Ontology engineering

Ontologies have become a most relevant representation formalism and many application domains rely on their adoption. Definitions can be used at least in two ways in the ontology building context.

The first usage of definitions is to describe concepts from an ontology. For the definition extraction, one can either start from the *definiendum* (concept) or from the

pattern of the definition. If the first option is taken, the automatic extraction procedure is similar to the QA context for definition questions, where the term is known in advance. The alternative is to start from the definition pattern. In this case, the situation is similar to the glossary creation context.

The second way in which definitions can be used in ontology building is for the extraction of semantic relations between the *definiendum* and the *definiens* (Storrer & Wellinghoff, 2006; Walter & Pinkal, 2006). Often, this semantic relation is expressed by the verbal phrase that connects the two parts of the definition. The verbal patterns used in the definitions for glossaries contain different types of such semantic relations. Apart from the "is a" relation, these are relations such as "consist of" and "used for". Ontology developers can profit from the definition extraction work by using these relations to automatically enrich ontologies. Both new concepts can be added and new relations can be expressed on the basis of the definitions extracted. However, although there is quite a number of semantic relations in the usage of definitions for the creation of glossaries, there may be still relevant relations that are not addressed (e.g. "is part of"). These could be included by extending the verbal patterns currently addressed by the glossary creation tool. From a machine-processing point of view, glossaries may thus be used as an input for domain ontology induction (see e.g. Bozzato *et al.* (2008)).

1.3 Challenges

Automatic definition extraction deals with several issues that are common to different Natural Language Processing tasks: inherent ambiguity (definition status is sometimes unclear), context dependency and data sparseness. We will check each of these aspects in turn.

Inherent ambiguity

As definitions support one of the basic functions of languages, it is a natural and easy task for humans to recognize them, but a complex activity for computers. Even if distinguishing definitions from other sentences should be an instinctive endeavor for us, if the concept of definition is not well or partially described, it can happen that some sentences are not consensually considered definitions. This means that sometimes, even humans themselves do not agree on the classification of whether a sentence is a definition

or not, because of different positions regarding what a definition should consist of, and how detailed or complete it should be.

We may be able to classify a sentence as being a definition if we can recognize that the sentence contains information that explains a term, even if we do not fully understand this information. For example, consider these four examples:

- *Acid rain is a rain with significantly increased acidity as a result of atmospheric pollution.*
- *Acid rain is a kind of rain.*
- *Acid rain is a problem with significantly increased consequences.*
- *Fridr is a voisder composed by three dertivds representing the minimal addressable element of a verfesd.*

The first sentence defines the concept of *acid rain* in a very clear way. The second give us some information about this concept, in particular its superordinate concept, without adding further relevant information, which however is enough to consider the sentence as a definition of the concept. The third sentence clearly does not offer a clear description of what the concepts is. In the fourth sentence, unreal words were used to show that even if we do not know the meaning of the key words of a sentence, and as a consequence, we do not understand the content of the definition, we may nevertheless recognize whether or not a sentence is a definition by looking at its structure.

Furthermore, our judgment on whether a sentence is a definition or not is influenced by our background knowledge and by our expectations. In the examples above, the second sentence can be considered an acceptable definition if you have a little knowledge of the subject and you do not need to deepen the concept of acidic rain too much. If you already have some knowledge about the concept, you probably need a more detailed definition. In this case, you will consider the second sentence only as a partial and not much useful definition. But if your domain knowledge is more structured, even the first sentence could not satisfy your need for defining the concept of acid rain. In this case you will need a definition more similar to the following: *Acid rain is a rain or any other form of precipitation that is unusually acidic (low pH), caused when sulfur dioxide and nitrogen oxides (from automobile exhausts and industrial emissions) are washed out from the atmosphere by rain as weak sulfuric and nitric acid.* On the other hand, a beginner could be confused by all the information contained in this definition.

As there are so many ways to define a concept, it is certainly a challenging task to identify the way humans classify definitions. We are able to classify a sentence as a definition if we recognize that the sentence contains information that explains a concept.

Context Dependency

The problem of context dependency and portability are due to the fact that there is a tangible worsening of performance when a system is tested with data that present some substantial difference from the data used for training. In recent years, the issue of lack of portability of Natural Language Processing (NLP) systems (such as part-of-speech tagging, semantic role labeling, statistical parsing, or machine translation) to new domains/genres of language started to get attention (Blitzer, 2008; McClosky *et al.*, 2006).

Regarding definition extraction, the context is given by the domain knowledge a given input text is about or related. A text can be characterized by having a specific type of language. For instance, the type of language used in a text of Psychology is different from the type of language used in one of Physics. Even for a given domain, the way to present and define a new concept can change taking into consideration the communication context if, for example, the text is a scientific paper to be read by experts or a tutorial for non experts.

A definitions extractor needs to take in consideration these peculiarities, as the problem here is how to build definitions extractors in a way that their performance does not get worse when used in different contexts of application, as pointed out by Malaise *et al.* (2004).

Data Sparseness

As for the last of the three issues to be checked, data sparseness, it represents a common issue in Natural Language Processing tasks given the Zipfian nature of many language phenomena and dimensions. The Zipfian law states that the probability of occurrence of words or other items starts high and tapers off. Thus, a few occur very often while many others occur rarely.

It is quite common to have to deal with a task where the positive examples of the phenomenon to be addressed are very few in comparison with the population. Examples

of this unevenly distribution are sentence boundary detection, word sense disambiguation and named entity recognition.

In the context of definition extraction, the distribution of the definitions in the text may vary from the one encountered in the body text of a dictionary to the one found in a literary text. In the first case, many sentences will be definitions; in the second case, it will be very unlikely to find a definition in the text. Apart from these two extreme cases, when we focus on domain-specific texts, we still get a certain degree of sparseness, which as we shall see in subsequent Chapters, can result in having a definition at around every 50 sentences on average. Clearly, the distribution changes with the nature of the text and its communication purpose. Again, a text addressed to experts in a given field will contain significantly less definitions than a text addressed to students in that area, for instance.

This unequal distribution has two major consequences. Firstly, it makes the observation of a large number of definitions difficult, as it would be necessary to collect and manually annotate corpora of very large size. Secondly, systems for automatic extraction tend to be very permissive in order to not miss the few definitions. This tends to result in a high number of sentences misclassified as definitions. This is especially evident when machine learning algorithms are used, as they present a bias towards the majority class (phrases that are not definitions).

1.4 State of The Art: Shortcomings

In this Section presents a brief overview of the weaknesses of the state of the art, that motivated the present research.

As it will be widely discussed in Chapter 3, the majority of the systems that automatically extract definitions have been constructed taking into account a specific corpus covering a specific domain knowledge.

This has a number of implications. Firstly, the types of definitions taken into consideration are the ones that come up most frequently in the text. For instance, depending on the structure of the text, one can have a large number of definitions that have the typical structure of the dictionary, that is, a definition where the term is defined at the beginning of the paragraph, often in bold, followed by a colon and then the description. A system focusing only on this type of definitions will get worse results when applied

to other documents. For example, Malaise *et al.* (2004) used two corpora of different domains, one to develop and another to test their system, with a worsening in performance when the system was applied to the test corpus. There are a few works of a more general nature, such as the one of Hearst (1992), that indicates some general patterns and proposes a heuristic to find new patterns for specific corpora.

Even in cases where texts do not have such a defined structure and corpora covering different domains are used, the selection of the type of the definitions to be taken into account is limited to a well-defined list of verbal and syntactic expressions with a low level of ambiguity. For instance, Alarcón *et al.* (2009) selected a list of verbal expression such as *is called*, *is defined*, etc. and then they constructed a set of rules to capture the definitions containing these verbal expressions. As the corpus used was not previously manually annotated with definitions, there is no way to know which other types of definition were present in the corpus. This type of study focus thus on an arbitrary and limited selection of verbal expressions, excluding all the other ways in which a definition can be expressed.

Most of the systems trying to deal with this task are based on the manual construction of a set of rules or patterns capable of identifing a definition in a text. Generally speaking, these studies share the assumption that automatic definition extraction is possible by looking for recurrent definitional patterns, either typographical or lexical. Typographical patterns refer to text typography or punctuation marks, while lexical patterns refer to syntactic patterns. We will describe further these patterns in Chapter 2 and 3, where applied investigations aiming at elaborating methodologies for the automatic extraction of definitional knowledge are discussed.

When handling unrestricted texts, it is difficult to determine in advance what kind of information will be encountered, and how it will be expressed. There are so many ways in which definitions are conveyed in natural language that it is difficult to come up with a closed set of linguistic patterns to solve the problem of definition extraction. To make matters even more complex, patterns are usually too broad, matching non-definitional contexts as well as definitional ones. In order to deal with this issue, some systems try to improve precision (reducing the number of sentence erroneously classified as definitions) by introducing the use of several predefined lists of words that are found in definitions. Even if it is possible to imagine a generic list composed by lexical items

such as *kind of*, *type of*, etc. that are domain independent, in practice its usefulness ends up being limited when dealing with a heterogeneous collection of texts.

A different solution for filtering misclassified definitions resorts to the introduction of machine learning algorithms, in order to recognize patterns to filter irrelevant candidates. The issue with this second solution is a worsening in the number of definitions correctly classified by the pattern matching module, as the filtering tends to discard some of the good candidates.

To summarize, excluding some very general heuristics, whenever one needs to build a system to extract definitions, one must start almost from the beginning, starting to analyze a possible set of example definitions and building *ad hoc* patterns.

1.5 Objective and Contributions

The challenge addressed by the present research work is to develop a set of methods and heuristics allowing to automatically distinguish definition sentences from non-definition ones.

We aim at providing a set of methods and heuristics for building a definition extractor with a broad range of application, that can be applied regardless of the corpus at stake and even the language.

In order to develop and test the ensuing architecture, a generic definitions extractor for the Portuguese language is built. Concomitantly, each heuristic is tested using corpora in some different languages, namely in English, and partially in Dutch.

In order to address the problem of the ambiguity of definitions, this research focuses on the three types of definitions described above in Section 1.1 characterized by the connector between the defined term and its description: copula definitions, punctuation definitions and verbal definitions.

The strategy adopted can be seen as a "divide and conquer" approach. Differently from other works representing the state of the art, specific heuristics were developed in order to deal with different definition types. Despite the fact that this type of definition characterization is not introduced for the first time by this research, the way it is exploited represents a novelty in the area. For instance, Westerhout & Monachesi (2007) use the definition classification here presented, but then they use the same rule-

based methodology for extracting all types of definitions, without really exploiting this classification.

In the present work we thus propose to use different methodology for each type of definition, namely we propose to use rule-based methods to extract punctuation definitions, machine learning with sampling algorithms for copula definitions, and machine learning with a method to increase the number of positive examples for verbal definitions. This architecture is justified by the increasing linguistic complexity that characterizes the different types of definitions. Numerous experiments have lead to the conclusion that the punctuation definitions are easily described using a set of rules. These rules can be easily adapted to the relevant context and translated into other languages. However, in order to deal with the other two definitions types, the exclusive use of rules is not enough to get good performance and they call for more advanced methods, in particular machine learning based ones.

The development of each one of the three approaches was determined by the goal of enhancing the portability of the final system, that is, rules, features, and other external resources were selected in order to be, as much as possible, domain and language independent. At the same time, the methods here developed are as much as possible semi-automatic, which means that they can be used to obtain a new extractor in a few automatic or semi automatic steps.

The methods proposed here are general enough to be applicable to different languages. This characteristic represents an innovation in the field of definitions extraction. Unlike other similar systems, which were built having in mind a specific corpus or a specific domain (Muresan & Klavans, 2002; Sánchez & Márquez, 2005; Walter & Pinkal, 2006), the one reported here is meant to obtain good results regardless the domain or context. All the decisions made in the construction of the definition extractor take into consideration this central objective.

Regarding the issue of data sparseness, this work proposes two different approaches to cope with it. First, a set of methods to semi-automatically extract definitions from corpora in order to increase the volume of data. Second, a combination of algorithms to sample the data set. Following this approach, the main contribution is to find which combination of machine learning algorithms produces the best classifier for extracting definitions. The novelty resides mostly in the combination of several sampling and learning algorithms for manipulating the data set, in order to overcome the data sparseness.

Furthermore, as little research has been carried out for the Portuguese language, an actual definition extractor for Portuguese represents a new resource for the development of NLP for this language.

1.6 Dissertation Outline

The first two chapters of this dissertation discuss in detail relevant research focused on definitions, addressing both theoretical and practical issues. Chapter 2 focuses on the discussion of what a definition is, giving a brief overview of related studies, from the Greek philosophers to the present day. Particular attention is given to the discussion of concepts and theories that underpin the possibility of extracting definitions automatically, such as the distinction between word and term, and between general and specialized language.

Chapter 3 starts describing the main differences between semantic relations extraction and definition extraction. Then, it proceeds by discussing the state of the art in the task of definition extraction, reporting on work carried out taking into consideration specific domains or applications such as glossary creation or question answering systems, as well as more general ones.

The following three Chapters describe the proposed method for each type of definition to be addressed. In Chapter 4, the two main corpora (Portuguese and English) used to carry out this research are presented. Then, the pattern based module for punctuation definitions is described. The development part was carried out using the corpus in Portuguese. Subsequently, rules were easily adapted to English. Results for both languages are presented and discussed.

Chapter 5 focus on copula definitions and on a learning based module using sampling algorithms. Firstly, a detailed discussion of the imbalanced data issue (arising when this kind of approach is used) is given, followed by the description of experiments, in terms of which learning and sampling algorithms were used. Then, results are presented and discussed.

Chapters 6 is devoted to verbal definitions handled using an external resource in combination with machine learning algorithms. In particular, Wikipedia was used to make the definition extraction more robust and to overcome the problem of having a great variability in definitions in together with a small number of examples.

Chapter 7 describes an implementation of the Portuguese definitions extractor in an e-learning environment. The system was tested by students and teachers using an evaluation methodology based on scenarios.

Finally, we conclude in Chapter 8 by presenting a discussion on the work carried out and observations that motivate future work in definition extraction.

Chapter 2

Background

2.1 Introduction

Before offering an overview of the research on automatic definitions extraction, we provide an overview of the studies on definitions, starting from the early philosophical reflections till the recent research conducted in the context of Terminology, that offers a theoretical background for the task of automatic definitions extraction.

This Chapter is divided into two parts. The first part offers an overview on the basic concepts supporting the automatic extraction of definitions. In particular, the analysis of definitions is based on theoretical constructs such as sublanguage and specialized language. The notion of sublanguage concerns the fact that various areas of knowledge are characterized by a specific type of language, presenting e.g. specific rules or lexicon that are determined by a specific communicative context and purpose.

In the last decades, due to the expansion of science, technology and communication a new need for clarification of new and old notions has emerged in order to eliminate possible ambiguities. In particular, this caused an increasing dissemination of information among non-specialists that induces the need to clarify and explain the meaning of terms used by specialists. For this reason, there is a growing interest in the development of systems for the automatic extraction of information that describe the meaning of terms occurring in specialized texts or in a specific knowledge field.

Much of the notions on which the automatic extraction of definitions is grounded derives from the field of Terminology. Terminology work is based on the identification of concepts that characterize a specific area of study. Technical concepts take their

linguistic form as terms and these terms are associated to definitions. The terminology work includes, after the identification of concepts and terms associated with them, the elaboration of definitions. This process can be done by consulting experts or by consulting texts. In recent years, the increasing availability of documents in electronic format and the advancement in the automatic information extraction make the automatic extraction of definitions more effective.

The second part of this Chapter proceeds with an account of what is a definition. In particular, it starts with a discussion about the origins of the notion of definition in a more theoretical perspective, corresponding to the origins of Philosophy itself. Initially, philosophers and logicians did not discuss the notion of definition as a textual element but as a method of knowledge. From this perspective, they were not interested in how definitions appeared in documents, instead they were interested in the properties that definitions were required to have in order to properly convey knowledge, and how different kinds of knowledge should be conveyed by different types of definitions. In this scope, a classification of definitions by purpose and method is provided. Finally, we give an overview of studies on definitions as these appear in naturally occurring documents, based on lexicographic and terminological approaches. Some characteristics and classification are, in most of the cases, derived from the early theoretical studies, for this reason there are some overlapping notions.

The assumption underlying the present work is that definitions constitute an individuated sublanguage themselves, with its own syntactic and semantic constraints, which emerge in a certain restricted knowledge (characterized by a specialized language) and in a certain communicative context.

2.2 Definitions as Sublanguage

Zellig Harris was one of the first to propose a theory of sublanguages that seeks to explain why it is possible to process language in specialized textual domains (Harris, 1968). By exploiting the notion of sublanguage, it is possible to discover units of information or knowledge and the relationships between them within existing knowledge sources, including published literature or corpora of narrative text. Several works have been conducted that explore the notion of sublanguage, in different areas such as Biology (Friedman *et al.*, 2002), Medicine (Sager *et al.*, 1994), or corporate information

(Symonenko *et al.*, 2006), etc.

According to Harris, all instances of a language are constituted by sequences of words following certain constraints. In particular, a general language can be distinguished from a sublanguage regarding two different type of constraints: dependency relations and difference of likelihood.

Dependency relations are concerned with syntactic regularities, such that the occurring of a word in a sentence depends on other words in the same sentence. Words that do not depend on other words in the sentence such as nouns or concrete objects are considered zero-level words (e.g. *cat*, *dog*). Words that depend on these zero-level words are called first-level words. Usually these words are verbs, which are considered to be operators that are dependent on their arguments (zero-level words). In the sentence *the dog chased the cat*, *dog* and *cat* are zero-level arguments, while the verb *chased* is a first-level operator. This language constraint is concerned with classes of words, not with individual words, and permits strange or unlikely combinations, such as *dogs chased computers*, as long as the dependency constraints are met.

Differences of likelihood are concerned with the fact that certain arguments are more likely to occur with certain operators than with others. For example, *dog* is more likely to occur as the first argument of *eat* than *computer*. The likelihood of an operator with respect to particular arguments is based on the frequency of operator-argument combinations. Some combinations occur frequently whereas others occur very rarely.

When comparing general language with a sublanguage, it is possible to observe that the sublanguage contains operators that are much more restrictive than the dependency relations permit in the general language and the likelihood constraints are much more definitive. For example, in a general language it is allowable, although uncommon, to say *David activated protein A*, because the syntactic combination of word classes is well-formed. However, in the sublanguage of Biology, this sentence is not legitimate because the operator, that is the verb *activate*, permits only certain combinations of the word classes (i.e., a substance may activate another substance, but it is not acceptable for a person to activate a substance), and the sublanguage operators reflect the relations and the arguments that are appropriate in the specialized domain.

In a specialized sublanguage, operators and arguments satisfy the dependency relations of the whole language, but the vocabulary is limited. Only restricted combinations

of words occur and subclasses of words combine in well specified ways with other subclasses.

Whereas a general language is characterized by well-formed syntactic structures, a sublanguage also incorporates domain-specific semantic information and relationships to delineate a language that is more informative because it reflects the subject matter and relations of a domain as well as the syntactic structure. For example, in the sublanguage of medical records, a speaker would accept the sentence *the X-ray revealed a tumor* but not *the tumor revealed an X-ray* (Grishman *et al.*, 1986).

As in most distinctions matters, there is not a cut off point between general and specific languages, but a continuum. For this reason, it is sometimes difficult to draw a clear line between a specialized language and the general language. However, there are certain factors that can help the understanding of the basic differences between the two. This distinction must take into account both the particular language used and the communicative context in which it is used. Individuals communicate differently in different situations and how they express their knowledge depends on both the context and situation and the kind of knowledge expressed (Pearson, 1998).

Barnbrook (2002) defines definitions as a sublanguage itself. In his study based on dictionary definitions, he describes the main characteristics of the language used in definitions that support his claim.

First, he reports the presence of a certain degree of lexical restriction. He compared a corpus composed by definitions extracted from an English dictionary with a corpus of the same size composed by newspaper articles. The result was that the number of words in the lexicon was 8,579 for the definition corpus and 27,814 for the newspaper corpus, and the number of words appearing just once was 2,501 and 12,107 for the two corpora respectively. This means that a definition corpus presents a restricted vocabulary in comparison with a general corpus.

Second, definitions are characterized by the occurrence of syntactic restrictions, such as, for example, the absence of interrogative and imperative sentences. Furthermore, sentences are organized following a well defined structure, composed by an identifiable *definiens* and *definiendum*, connected by a verb. The connector verb is represented most of the time by the verb *to be* or *to mean* or its synonyms. Moreover, it appears in the third person (singular or plural), in the present tense. As definitions normally describe current meanings of currently used words, only in few cases the connector verb

appears in a past tense. Finally, definitions only rarely exhibit the presence of pronouns referring to expressions present in previous sentences.

Third, definitions also present semantic restrictions. Words performing structural functions such as *people, person, thing, place*, etc., that have several senses in a general corpus, in a definition corpus are used mostly in their more common sense. For instance, the word *people* present several meanings such as *a human being*, *a human body*, *a grammatical category*, *a character or part, as in a play*, *a shoot or bud of a plant*. However, when this word appear in a definition, it is used in its more common meaning, that is it means *a human being*.

Finally, definitions exhibit a high frequency of certain constructions. Barnbrook (2002) identifies seventeen different structural patterns, where the first eight more frequent patterns account for over 92% of all the definitions. All these observations support the claim that the language used in definition bearing sentences can be considered a sublanguage.

If we use a general framework to analyze definitions, we will end up with a lot of information regarding a sentence that is not useful with respect to its communication purpose. To analyze general language is very hard because a grammar dealing with general language should take into account all the communication possibilities of that language. If we focus on a sublanguage, we restrict our field of inquiry to a more circumscripted one. In the case of definitions, we have to deal just with one communication purpose, that is to provide information describing the meaning of a term. A model describing definitions does not need to deal with every grammatical aspect of a definition bearing sentence but only with few components such as the *definiens*, the connector and the *definiendum*. Starting from these basic components, it is possible to refine the model, including the identification of more components in the *definiens*, such as superordinate concept, field or discriminators.

2.2.1 Specialized Languages and Terms

In the last fifty years, we have witnessed an unprecedented development in science and technology followed by an increasing interdisciplinarity requiring uniformization of the expressions associated to concepts that belong to various disciplines. The transfer of knowledge and products, considered one of the most important aspects of modern society, leads to the emergence of new markets, to the exchange of scientific, technical,

cultural and commercial information that requires to set up and solve the multilingualism issue and, finally, causes the need of standardization. This sudden increase of information requires the development of systematic and organized terminology, that is, the clarification of concepts and standardization of nomenclature to facilitate communication. In this context, Terminology emerged as a discipline.

Terminology encompasses two major notions: specialized language and term.

As there is a distinction between general knowledge and expert knowledge, there is also a difference between the language used in general communicative situations and the language used in the communication of knowledge related to a specific area. The language used in communicative situations, whose purpose is the transmission of knowledge shared by members of a group, is known as specialized language. A specialized language is a sublanguage used in a particular knowledge domain.

After discussing the difference between general language and sublanguage, it is important to clarify the difference between word and term. As a matter of fact, it is quite difficult to understand the former notions without the latter ones as they are interconnected in such a way that one can say that the notion of general language aligns with the notion of word, and specific language aligns with term.

While a word is a unit of general language, a term is a linguistic sign within a specific domain of discourse. Typically, a word may have several senses. A term, in turn, is a word used in a precise discourse context and bearing one precise sense in that context. Terms are not, by definition, part of the general linguistic competence of a native speaker of a language. The meaning of terms is formed by agreement among the expert in a specific knowledge field, for this reason it is highly conventional, even if they were derived from words that originally belonged to the general lexicon.

Terminology is the study and classification of terms associated with a particular field in order to systematize the concepts related to a specific subject area. Terminological work aims to identify knowledge about terms in specialized texts in order to compile dictionaries, glossaries or ontologies (ISO 704, 2009).

The objectives and methodologies of this discipline can be briefly described as:

- identifying concepts and concept relations;
- analyzing and modeling systems of concepts on the basis of identified concepts and concept relations;

- establishing representations of concept systems through concept diagrams;
- defining concepts;
- attributing designations (predominantly terms) to each concept in one or more languages;
- recording and presenting terminological data, principally in print and electronic media (terminography).

Concepts constitute the starting point of this terminology work as they are considered units of thought and knowledge. They are in correspondence with objects or sets of objects in a specific subject area and are expressed in language by terms and clarified by definitions. Objects are identified by their properties. They are abstracted as concepts and the properties are abstracted as characteristics making up the concepts. The set of characteristics that form the concept is called the intension of the concept. The set of objects that correspond to a concept is known as the extension of the concept. The two, intension and extension, are interdependent.

Concepts do not exist as isolated units of knowledge but are organized into concept systems through semantic relations, such as hierarchical relations (generic relations and partitive relations) and associative relations. The term is considered the minimum unit of the terminology.More specifically, in a specialized language, the term is a designation for a concept (ISO 1087, 2000).

Both the term and the definition represent the concept. This means that the concept, the term and the definition all relate to the same object or objects. A term is a succinct way of naming a concept, while a definition is a descriptive statement that differentiates the concept from others within the relevant domain.

As there is a distinction between term and word, there is also a distinction between the definition corresponding to a term or to a word. While a definition in lexicography describes the meanings of words indicating its possible senses, a terminological definition defines the meaning of a term with reference to the conceptual system to which it belongs. The elaboration of the definition is an important step in terminology, which aims at the elaboration of specialized dictionaries. A brief description of the terminological definition is given in Section 2.3.2.

2.2.2 Where Definitions are Found

The work on definitions in terminology is based on domain corpora characterized by the use of a specialized language. It is possible to focus a bit more on this issue and analyze the documents used, taking in consideration the intended final users. As pointed out by Aristotle in *Topics* (Aristotle, ca. 350 BCb), "Different things are more intelligible to different people, not the same things to all, and so a different definition would have to be rendered to each person, if the definition is to be constructed from what is more intelligible to particular individuals".[1]

Taking into account that each discourse domain may have a set of lexical items that differentiates it from the other domains, each communicative situation may use more or less specific vocabulary. Accordingly, there will be more or less need to explain that lexical set depending on the communicative situation: while the communication among experts will assume that many of the concepts do not need any explanation, the communication with the semi-skilled persons and non-experts will be as much as possible explanatory, repeatedly using definitions which clarify the meaning of the particular vocabulary used.

Pearson (1998) indicates that the communicative situations that involve different written or oral expressions of specialized language can be classified into three types:

- Communication between experts
- Communication between experts and beginners
- Communication between experts and non-experts

In the first case, the language used in the communication is highly specialized as the source and the target of the communication have the same level of expertise, share a common background in terms of knowledge and use highly specific terms, which in most cases are defined before the communication happens, probably by an external authority. In this context, definitions are provided only in the case a term is used in a different sense from the one established or when new terms are added in the field. This type of communication is typical in journals, academic books, etc.

[1] *Topics*, Book VI, 139a-151b, 4.

In the second case, the source and the target do not share the same level of expertise on the given subject, but subject-specific knowledge is involved. The communication producer has a deeper knowledge on the matter than the receiver. In this case, one finds many definitions being used, which tend to be clear, detailed and specific. This type of setting is found, for instance, in textbooks.

In the last setting, no subject-specific knowledge is assumed for the target of the communication. The non-experts are those who are involved in the communicative situations in a given field, and general language is used as much as possible. Definitions tend thus to be abundant, but less detailed and specific. This setting is common in publications aiming at scientific dissemination.

2.3 Classification of Definitions

The word *definition* is actually used to denote different situations involving the human activity of clarifying a concept. It may refer to a sentence uttered or imagined, or just to a part of the sentence, or also to the meaning of the sentence. In order to deal with definitions in the area of Natural Language Processing, what has been discussed in the past about the classification, properties and characteristics of definitions should be taken into consideration. Barnbrook (2002) pointed that "definitions form one of the basic functions of languages, and a description of their structure and operation describes a major aspect of language use". This characteristic is probably what motivates a large investigation on definitions and, at the same time, what makes this term so ubiquitous in philosophy since Antiquity.

The discussion about a typology for definitions is as old as the discussion about definitions in general. Here we will present only the most significant classifications.

When overviewing the notion of definition in the context of the early philosophical discussion, a first major distinction should be referred, that is the distinction between the so called **real** and **nominal** definitions (Robinson, 1950). The former are associated with the first philosophical reflections on the concept of definition and deal with definition as an ontological method. The latter, which emerged much later during the seventeenth century, handles definitions as a means to report or to determine the meaning of a symbol.

A real definition aims at explaining the nature of a concept. It goes beyond providing awareness that something exists, aiming at telling us what it is. A nominal definition explains the meaning of a word or a term. For example, the word *thunder* could be defined as *a noise in the clouds*. This gives enough information to know what the word *thunder* is referring to, but it does not tell us much about what a thunder really is. This division into nominal and real definitions characterizes the philosophical debate for long time.

When in the 4th century BC, Plato introduced what we would see nowadays as the notion of definition, he only thought of definitions of things and not of words or symbols. In one of his dialogues, Plato (ca. 360 BC) shows Socrates affirming that "right opinion with rational definition or explanation is the most perfect form of knowledge".[1] According to Plato, a definition represents the culmination of the process of getting to know a thing. Furthermore, he pointed out in which way a definition could fulfill its task, stating that "if you get at the difference and distinguishing characteristic of each thing, then you will get at the definition or explanation of it".[2] In *Topics*, Aristotle (ca. 350 BCb), clearly states that "... the definition consists of *genus* ($\gamma\acute{\epsilon}\nu o\sigma$) and *diferentiae* ($\delta\alpha\phi o\rho\acute{\alpha}$)".[3] This way of offering a definition was considered by Aristotle as the most appropriate method for reasoning and presenting the results of analysis. For this philosopher when we have a true account of the essence of a thing then we have the most important possible knowledge. In *Posterior Analytics*, Aristotle (ca. 350 BCa) claims that a "definition is held to concern essential nature and is in every case universal and affirmative".[4] Again, it is a process not about words, but about things and reality, and how we get to know reality.

In summary, greek philosophers cared especially about the existence of things and the causes for this existence, and used definition as a method for reaching knowledge about reality. This approach assuming definition as a way leading to knowledge has been common ever since. Even in the seventeenth century, Spinoza (1677) wrote "the true definition of each thing involves nothing and expresses nothing but the nature of the thing defined".[5]

[1] *Theaetetus*, 206.
[2] *Theaetetus*, 208.
[3] *Topics*, Book I, 101b-103b, 8.
[4] *Posterior Analytics*, Book II, 90a-98a, 3.
[5] *Ethics*, I, prop. 8, n.2.

In the same century, John Locke adopted an account of definitions as regarding names and not only things (Lock, 1690). He drew a distinction between real and nominal definitions, offering an example between the two, using the word *gold*. *Gold* can be defined as "the constitution of the insensible parts of that body, on which those qualities and all the other properties of gold depend"[1] or as "that complex idea of the word gold stands for..., a body yellow, of a certain weight, malleable, fusible and fixed".[2] The first is an example of real definition, as it states real essence, while the second is an example of nominal definition, as it states nominal essence.

The discussion on real and nominal definitions continues till recent days. Gupta (2012) suggests that the use of real and nominal definitions depends on the purpose and circumstances in which a term is used, so that there is not a clear distinction between these two types of definitions. For example, a zoologist's definition of *tiger* should count as a real definition, even though it may fail to provide the real essence of the tiger, while an account of the meaning of a word should count as a nominal definition, even though it may not take the Lockean form of setting out "the abstract idea to which the name is annexed".[3] Thus, Gupta proposed a method to discriminate between real and nominal definitions. In order to find out the real definition of a term X, one needs to investigate the thing or things denoted by X; while in order to find out the nominal definition, one needs to investigate the meaning and use of X. In this way the purpose of the definition determines whether the definition is real or nominal. Even when Socrates asks for the definition of *virtue*, the answer can result in a real definition if he is trying to give an account of an ideal that is to some extent independent of these uses, or it can result in a nominal definition if he is trying to gain a clearer view of our uses of the word *virtue*.

Even if the purpose is different, the linguistic concretization of the definition in the text is the same. This means that the automatic definition extraction is not affected by this differentiation. Nevertheless, nowadays the term *definition* generally refers to nominal definitions, since usually the aim is to find out what a word means when one uses it.

[1] *Essay concerning Human Understanding*, Book I, Ch. 4, 2.
[2] *Essay concerning Human Understanding*, Book IV, Ch. 6, 8.
[3] *Essay concerning Human Understanding*, Book III, Ch. 6, 2.

2.3.1 Classification by Purpose

Along with the division of definitions into real and nominal, different classifications into subtypes have been proposed. Robinson (1950) inserted the distinction between real and nominal definitions in the broader context of the classification of definitions. In particular, definitions can be analyzed according to their purposes and according to the method by which they are expressed.

The purpose is what one tries to achieve with a definition. Hence, the purpose of a real definition is to tell us what a thing is, while the purpose of a nominal definition explains the meaning of a word or term.

Focusing on nominal definitions, Robinson (1950) identifies four types of nominal definitions: word-word definitions, word-thing definitions, lexical and stipulative definitions. Figure 2.1 shows the complete classification given by this author.

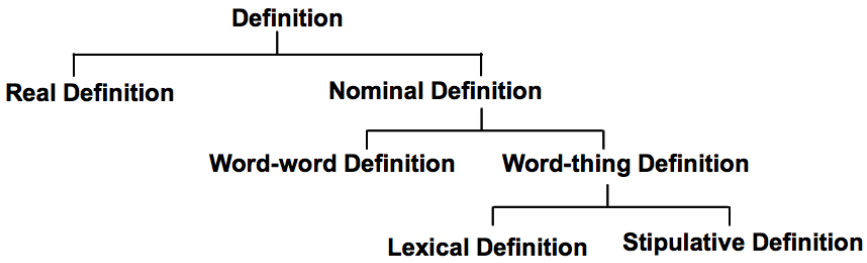

Figure 2.1: Definition classification proposed by Robinson (1950)

A **word-word definition** connects two words with the same meaning, with no reference to the real word. This kind of definition states the equivalence between two symbols, as in the example *the symbol "=" means "equal to"*.

A **word-thing definition** aims to report or establish the meaning of a symbol. In Robinson's view, words are means by which humans deal with things, so a definition is about words, things and humans. This kind of definition not only reports on a rule about words and things, but also can grant new knowledge. For instance, if someone asks for the meaning of *compound fracture* not only he can learn about the expression *compound fracture*, but he can realize for the first time the existence of such thing. Robinson distinguishes two types of word-thing definitions: lexical and stipulative definitions.

A **lexical definition** is a statement of the existence of a rule by which a certain form was used as a sign of a certain thing. This type of definition reports the way in which a term is already used within a language community. The goal here is to inform someone else of the accepted meaning of the term, so the definition is more or less correct depending upon the accuracy with which it captures that usage. Definitions in a dictionary are examples of lexical definitions, as dictionaries aim to provide definitions that contain sufficient information to offer an understanding of the term.

A **stipulative definition** is a proposal or request that there shall be a rule by which a certain linguistic form was used as a sign of a certain thing. It assigns meaning to an expression, creating an usage that had never existed previously. Since the goal in this case is to propose the adoption of a shared use of a novel term, there are no existing standards against which to compare it, and the definition is always correct (though it might fail to win acceptance). This type of definition serves the specific purposes of domain specific experts, such as legislators, law firms, commercial firms, etc. Stipulative definitions usually apply only within the parent document or set of related documents to which they apply. A famous example in literature is this passage of *Through the Looking-Glass* by Lewis Carroll

> - I don't know what you mean by "glory", - Alice said.
>
> Humpty Dumpty smiled contemptuously. - Of course you don't, till I tell you. I meant "there's a nice knock-down argument for you!" -
>
> - But "glory" doesn't mean "a nice knock-down argument", - Alice objected.
>
> - When I use a word, - Humpty Dumpty said, in rather a scornful tone, - it means just what I choose it to mean, neither more nor less. -
> (Lewis Carroll, *Through the Looking-Glass*, 1871)

Logic (see, for instance, (Copi & Cohen, 1990; Hurley, 2011)) gave a big contribution to the study of definition typology, distinguishing several other definitions types. Here we report on three more types: precising, theoretical and persuasive definitions.

Precising definitions aim to sharpen a lexical definition by stipulating more narrow limits on its use. Here, the lexical part must be correct and the stipulative portion should appropriately reduce the troublesome vagueness. This type of definition may be used for adapting a definition to a specific subject field. It may begin with a lexical

definition and may involve turning the lexical definition into a precising definition, i.e. giving a more ample lexical expression into a definition with more precise characteristics. The objective is to identify the specific concept designated by a specialized technical or scientific term. This is similar to a stipulative definition but differs in that a stipulative definition may contradict the lexical definition, while a precising definition does not. Furthermore, while stipulative definitions can be completely arbitrary, precising definitions are limited by the more general lexical definition. Consider a situation where two people are arguing whether animals such as birds or apes possess language. To resolve this dispute, we need to be more precise as to what is meant by *language*. If by *language* we refer to any system of communication, then obviously birds and other animals do make use of language. On the other hand, *language* might be used in a different sense, requiring a combinatorial syntax and semantics, allowing a user of the language to communicate information about non-existent objects or situations remote in time and space from the location of discourse. Used in such a way, the communication system of some animals might not qualify as a language. This example illustrates the use of precising definitions to resolve disputes that involve some key concepts whose meanings might not be clear enough.

Theoretical definitions are a special case of stipulative or precising definition, individuated by their attempt to establish the use of this term within the context of a broader intellectual framework. Since the adoption of any theoretical definition commits us to the acceptance of the theory of which it is an integral part, we are rightly cautious in agreeing to it. Newton's definition of the terms *mass* and *inertia* carried with them a commitment to (at least part of) his theories about the conditions in which physical objects move.

Persuasive definitions are an attempt to attach emotive meaning to the use of a term with the aim to influence the attitude of the reader. The focus of this type of definition is neither true or falsity, but the effectiveness as instrument of persuasion. For this, they are used mostly in political speech and editorial columns. The sentence *Taxation means the procedure used by bureaucrats to rip off the people who elected them* in an example of persuasive definition.

2.3.2 Classification by Method

Over time, logicians along the time also distinguished several methods to provide a definition, that are linked with the distinction between the extension and the intention of a concept (Copi & Cohen, 1990; Hurley, 2011). The extension of a general term is just the collection of individual things which it correctly identifies. Thus, the extension of the word *chair* includes every chair in the world. The intension of a general term, on the other hand, is the set of features which are shared by everything to which it applies. Thus, the intension of the word *chair* is something like "a piece of furniture designed to be sat upon by one person at a time."

Clearly, these two dimensions of meaning are closely interrelated. We usually suppose that the intension of a concept or term determines its extension, that is we decide whether or not each newly-encountered piece of furniture belongs among the chairs by seeing whether or not it has the relevant features. Therefore, as the intension of a general term is more detailed, by specifying with greater detail those features that a thing must have in order for it to apply, the term's extension tends to decrease, since fewer items now qualify for its application. Figure 2.2 presents an overview of the classification by method.

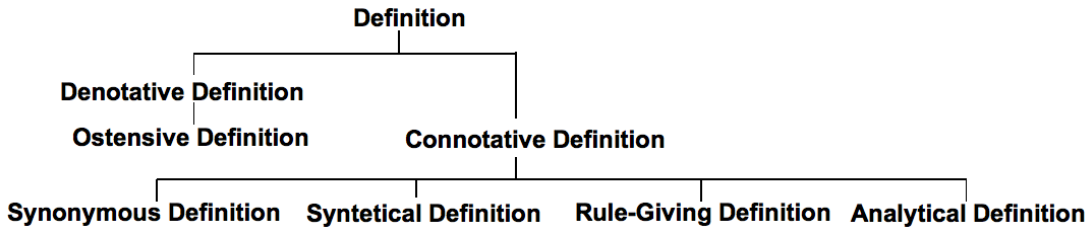

Figure 2.2: Definition classification by method

Regarding the extension, **denotative definition** is a method in which the meaning of a word is conveyed by citing examples taken from the class of objects to which the word is applied. For example, to define the word *ocean*, one could list the Atlantic, Pacific, Indian, Arctic, and Antarctic bodies of water.

The most primitive example of denotative definition involve no more than pointing at a single example to which the term properly applies. This type of definition is called **Ostensive definition** and consists in pointing to the object (regardless if the object

is present or not), not relying only on word. For instance, *Redfigure vases are the sort you saw yesterday in my study* or *A caterpillar is that animal* - while one points point to the animal.

Instead of pointing at a concrete example, it is possible to enumerate the members of the class the term denote or the member of the subclass. For instance, *Mediterranean country means France, Italy, Spain, Greece* or *Fruit means apple, banana, orange, etc.*

A serious limitation with denotative definitions is that it is often not viable to enumerate all members of the class. Since a complete enumeration of the things to which a general term applies would be cumbersome or inconvenient in many cases, we commonly pursue the same goal by listing smaller groups of individuals or by offering a few examples instead. However, there seems to be some important terms for which denotative definition is entirely impossible. The phrase "my grandchildren" makes perfect sense, for example, but since it presently has no extension, there is no way to indicate its denotation by enumeration, example, or ostension. In order to define terms of this sort at all, and in order to define general terms of every variety more conveniently, we naturally rely upon the second mode of definition. However, most phenomena have attributes that could place them in the extensions of many different terms. Thus, the use of this method can leave the meaning of the term we wish to define uncertain.

Regarding the intension of a concept, **connotative definitions** aim to identify the intension of a term by providing a synonymous linguistic expression (synonymous definition) or an operational procedure for determining the applicability of the term. Returning to our example, one possible definition of the word *ocean* would be "those bodies of salt water with an area greater than five million square miles". Of course, it isn't always easy to come up with an alternative word or phrase that has exactly the same meaning or to specify a concrete test for applicability. But when it does work, connotative definition provides an adequate mean for providing the meaning of a term.

A good connotative definition should follow five rules in order to correctly provide the explication of the term defined (Copi & Cohen, 1990). First, it should focus on essential features, in oder to convey the essential meaning of the term. For example, a definition of *human beings* as "featherless bipeds" fails to provide the essential meaning, as it says nothing about the capacity of reason and the use of language, that are important features that distinguish humans from other animals. Second, definitions

should avoid circularity, as in the sentence *Science means the activity engaged by scientists*. Third, a definition should be neither too broad nor to narrow. For example, the definition of *bird* as "warm-blooded animal" is too broad, since that would include also horses, and dogs; while the definition of *bird* as "feathered egg-laying animal" is too narrow, since it excludes those birds who happen to be male. Fourth, it should avoid obscure or figurative language, for example as the sentence *History is the unfolding of miscalculation*. Finally, a definition should be, whenever possible, affirmative instead of negative. For example, defining *Concord* as *harmony* should be preferred to defining *Concord* as *the absence of discord*.

Synonymous definition represents the simplest method to provide a connotative definition. It consist in providing another word having the same general sense as the definiendum and with which the learner is already familiar. This is the most common method of definition used in dictionaries. For example, the words *opponent*, *antagonist*, *enemy*, and *foe* may be provided as synonyms for the word *adversary*. The principal advantage of this method is that it is concise and straightforward. Its disadvantage is that it assumes prior understanding of the meanings of the words provided as synonyms. Moreover, synonyms are broadly similar rather than exactly alike in meaning. They carry different shades of meaning that can mislead the learner. According to Robinson (1950), dictionaries often list several partial synonyms "in the hope that the false in each will be cancelled by the others".

Besides synonymous definitions, there are several other ways to indicate the intension of a term.

Synthetical definition consists in indicating what is meant by mentioning its relations to some other known things, or mentioning how it is caused or at which conditions it arises. For instance, *Feuillemorte is the color of withered leaves in autumn*.

Implicative definitions consist in giving a sentence which implies that the word means so and so. For instance, in the sentence *a square has two diagonals, and each of them divides the square into two right-angled isosceles triangles* the word *diagonal* is not explicitly defined, nevertheless if a person knows the meaning of all the other words in the sentence, he can infer the meaning of *diagonal*.

Rule-giving definitions consist in giving the rule of how a word is used. For instance, *the word I is to be used by each utter to indicate himself*.

Despite this rich classification and analysis, logicians focused most of their effort on a specific method of constructing connotative definitions, that is analytical definitions by *genus* and *differentia*. **Analytical definition** is a method in which the phenomenon with which the meaning of the term is connected is broken down into its constituent elements. The basic notion is simple: we begin by identifying a familiar, broad category or kind (the *genus*) to which everything our term signifies (along with things of other sorts) belongs; then we specify the distinctive features (the *differentiae*) that set them apart from all the other things of this kind.

The advantage of this method is that it not only conveys the meaning of the word, but also gives an analysis of the characteristics of the phenomenon itself. The disadvantage is that it is a more involved method of definition, certainly when compared with the simplicity of the synonymous method. For example, it is easier to define *adversary* as *an enemy* than to set out the attributes (not all of which may be generally agreed) that distinguish an adversary. Aristotle's definition (*genus* plus *differentiae*) is an example of this kind of definitory method. For instance, *the word octagon means a polygon having eight sides.*

Copi & Cohen (1990) affirm that this definition is superior to any other definition, and that the *genus* and *differentia* method is usually the "most effective and most helpful" of all definition methods. Some go even further, maintaining that the analytical method is, indeed, the only acceptable method. Mill (1843) wrote: ".. name, whether concrete or abstract, admits of definition, provided we are able to analyze, that is to distinguish it into parts, the attribute or set of attributes which constitutes the meaning both of the name and of the corresponding abstract".[1]

Recently, definition has been also the object of ISO standardization (ISO 1087, 2000; ISO 704, 2009). In the framework of terminology, six main types of definitions are distinguished: intensional, extensional, ostensive, lexical, precising and stipulative, that correspond to the definitions already described. They only differ with regards to intensional and lexical definitions: an intensional definition corresponds to the Aristotle formal definition, that is, includes a superordinate concept immediately above the defined one, followed by the delimiting characteristics; a lexical definition is the one found in a general language dictionary when the superordinate concept is not specialized,

[1] *A System of Logic Ratiocinative and Inductive.* Book I, Ch.VIII.

that is when the superordinate concept is constituted by words performing structural functions such as *people, person, thing, place.*

Studies presented so far deal with definition as it should be, more than as it appears in naturally occurring texts, representing good guidelines for writers when they introduce definitions in their texts.

2.3.3 Definitions in Context

The studies presented in this Section are concerned with definitions as they happen to occur in actual contexts of usage. This means that the classification and observations here reported are related to definitions as they occur in texts and that we want to automatically extract. Most of these studies use as documentary sources domain specific corpora to conduct preliminary studies for automatic definition extraction systems. It is not surprising if many of the notions presented in the previous Section can be found with some qualification in this Section. Along the centuries, the notion of an ideal definition has influenced definitions in their concrete manifestation in texts and vice versa.

The focus on analytical definitions impacts the study on automatic definition extraction and also other domains, such as lexicography, terminology and automatic definition extraction. In her study on terminography, following the Aristotelian characterization, Meyer (2001) describes two types of definitions that can be considered typical in texts, the formal and the semi-formal ones.

Formal definitions should resemble an equation, following the schema $X = Y + C$, where X is the *definiendum* (what is to be defined), " = " is the equivalence relation expressed by some connector, and the expression $Y + C$ is the *definiens* (the part which is doing the defining). The *definiens* should consist in two parts: Y expresses the *genus* (the nearest supeordinate concept), denoting the class of which the denotation of X is an instance or a subclass, and C expresses the *differentiae specificae* (the distinguishing characteristics) that turn the denotation of X distinguishable from other instances or subclasses of the denotation of Y. For instance, *the acid rain is a rain with significantly increased acidity as a result of atmospheric pollution.*

Semi-formal definitions only present a list of characteristics, while the *genus* is omitted. For example, *a canister is usually round and you put things in it.*

Gernot (2007), in his studies on definitions in translation, enumerates two more types of definitions besides the Aristotelian ones. Firstly, the definition by enumeration of the type of elements in the denotation of the concept (**extensional definition**), e.g. *chess piece is a king, a queen, a bishop, a knight, a rook or a pawn.* Secondly, the definition by enumeration of the individuals that instantiate the concept (**partitive definition**), e.g. *the solar system is made of the planets Mercury, Venus, Earth, Mars, Jupiter, Saturn, Uranus, Neptune and Pluto.*

In the same way, Sierra *et al.* (2006), analyzing a corpus rich in definitions, identify four types of definitions from the Aristotelian formal definition, that is:

- Exclusive *genus* definition: provides no description of the *differentiae.*
- Synonymic definition: indicates a strong semantic relationship with the *genus.*
- Functional definition: includes *differentia* that indicates the function of the concept.
- Extensional definition: includes *differentia* that enumerates the parts of the concept.

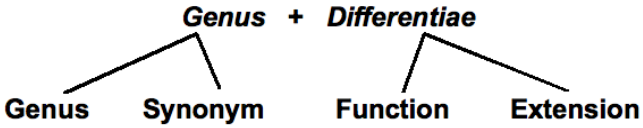

Figure 2.3: Aristotelian formal definition identified by Sierra *et al.* (2006)

In the first type of definition, the exclusive *genus* definition, only the *genus* is given. For instance, the concept expressed by the word *Java* is explained saying that is a programming language. Regarding the synonymic definition, in this case only a synonym is given. For instance, the concept expressed by the word *feature* can be also expressed by the words *characteristic* or *attribute.* We can, thus, conceive the meaning of a word by the meaning of another one that can replace it, and, in the present example, we understand its meaning by saying that *a feature is an attribute of an object.* In functional definitions, the *differentia* describes how to use or the purpose of the *definiendum*, as in

the sentence *a chair supports the weight of one person.* Finally, the extensional definition type incorporates the extensional and partitive definitions described by Gernot (2007).

In the studies on typologies of definitions occurring in texts, the notion of semantic relations plays an important role. At the same time, there emerges the observation that the template for analytical definitions is not sufficient for describing every possibility to formulate a definition in natural language (Sanger & Ndi-Kimbi, 1995). Along this line, Pearson (1998) introduces the notion of **non-formal definition**. This type of definition does not conform to a specific scheme, since it may have multiple configurations using both linguistic (verbal phrases, adjectives, adverbs, etc.) and non-linguistic elements (for example, typographical marks). She observes that when authors define a term, they usually use typographical patterns to visually highlight the presence of terms and/or definitions as well as lexical and metalinguistic patterns to connect them with their definitions by means of syntactic structures.

The semantic relations are present in the studies from the terminological typology of hierarchical relations inaugurated by Wüster *et al.* (1998), which until today is regarded as a reference in the area. Following this author, these relations can be of two types:

- Generic relation (*genus*-species) that can be described by "type of" using formulas:
  X is a type of A
  X, Y and Z are subtypes of A
  A contains X, Y and Z
  A contains the subtype X

- Partitive relation (part-whole) that indicate linkage between denotation of the concepts, that consists of more than one part and their constituent parts. It typically occurs associated to formulas such as:
  X is a component of A,
  X, Y and Z are components of A
  A consists of X

As a part of a multilingual research project named Aquilex, Copestake (1993) uses the semantic operator IS-A to delineate a set of taxonomies for classifying lexical definitions. Basically, she proposes three types of semantic relationships:

- Hyponymy-Hyperonymy: a hyponymic entity is related to a hyperonym, for example *an autobiography IS-A book*.

- Synonymy: two entities, that maintain certain equivalence are related, for example *a policewoman IS-A female policeman*.

- Individuation: relation between entities where a shift of individuation takes place; there are two kinds of individuation: a) quantity/mass, i.e. a relationship between a portion or a piece and a certain substance or entity, e.g. *an hour IS-A portion of time*; b) member/group, which is a relationship between an entity and a group or collective, e.g. *a policeman IS-A member of a police force*.

This list of relations, although relevant, is still incomplete. This circumstance led authors such as Aussenac-Gilles & Jacques (2006) to consider also the relations called non-hierarchical or complex, which requires a definition structure different from the generic and partitive ones. These relations can indicate a material, a process, a place or an activity and can be introduced by expressions such as *is caused by*, *is the place for*, *is the instrument for*.

The research in the field of automatic extraction of semantic relations from corpora, both for the generation of ontologies or for the extraction of definitions, reveal that there are four categories of very productive relationships, namely: hyperonimical, meronymic, causal and purpose. We have already illustrate hyperonimical relations above.

Meronymic relation expresses a link between the whole and its parts and is similar to the extensional definition described by Sierra *et al.* (2006), while relation of purpose, expressing the utility of an entity, corresponds to functional definitions that expresses the utility of an entity. Causal relation, in turn, links a cause to its effect, such as *X causes Y*.

Condamines & Rebeyrolle (2001) show another kind of non-analytical definitions used in discursive contexts. These authors automatically identify candidate terms and then analyze patterns contained in the sentences where the terms appear. In this way, it is possible to obtain yet other types of semantic relations. For example, in the sentence *the component development cycle takes place during the product realization phase* it is possible to identify a semantic relation of trajectory, where the phrase *takes place during* supposes a temporal frame that delimits the beginning and the end of an action.

The above studies indicate thus that there can be different types of definitions that are based on the analytical one but are characterized by several semantic relations. Furthermore, the extraction of definitional knowledge represents an interesting approach for the extraction of semantic relations. This idea was reinforced by Meyer (2001), who stated that definitional patterns can also provide keys that allow the identification of the type of definition found in discursive contexts which is a helpful tool in the development of ontologies.

2.4 Conclusions

In this Chapter we have addressed the study of definition from different angles, ranging from the philosophical view to the terminological studies, and including the logical approach. Despite a more fine grained classification in theoretical studies, when dealing with definitions in real contexts of usage, the superficial concretization of definitions and the semantic relations involved need to be taken into account. Lexical and typographical patterns should also be taken into consideration in order to characterize and extract definitions.

The creation of such patterns is made more achievable in the light of the concept of sublanguage. Specialized knowledge is associated with the activities and the communicative situations of a specific area, and is in contrast with the general knowledge that is present in a wider range of situations shared by members of a larger community. The language used in these communicative situations is a specialized language or sublanguage, that may be analyzed more effectively than general language, as it is more restricted. The act of defining can be considered a sublanguage, amenable to be analyzed as supporting, this way, the automatic extraction of definitions.

Chapter 3

State of The Art

3.1 Introduction

This Chapter presents an overview of several methods and systems that have addressed the issue of the automatic extraction of definitions.

The availability of huge volumes of information in electronic format has led to the development of various computational tools to facilitate information processing and extraction. Natural Language Processing (NLP) deals with issues related to the development of systems for the generation and understanding of human language. In particular, some areas within the NLP have been directed specifically to process textual information with the purpose of developing systems for search and selection of documents that meet certain criteria outlined by a user. As examples we have Internet search engines, or systems for the search and selection of specific data on events, entities or relationships from a set of documents. NLP has focused, among others, on the extraction of terminological information and also definitional knowledge, that is information that allows capturing the meaning of the terms from the description of its attributes, characteristics or semantic relationships.

Systems for the detection and extraction of definitions have been developed for different purposes. In particular, several systems were developed in the last years in order to create glossaries (Muresan & Klavans, 2002; Park *et al.*, 2002), lexical databases (Alshawi, 1987; Nakamura & Nagao, 1988), ontologies (Baneyx *et al.*, 2005; de Freitas, 2007; Walter & Pinkal, 2006), question answering tools (Androutsopoulos & Galanis, 2005; Chang & Zheng, 2007; Saggion, 2004; Tjong *et al.*, 2005), or to support terminology applications (Meyer, 2001; Seppälä, 2009). Most of the systems trying to deal with

this task are based on the manual construction of a set of rules or patterns capable of identifing a definition in a text. In a few cases, statistical or machine learning techniques are also used to improve the results.

When interpreting unrestricted, domain-independent texts, it is difficult to determine in advance what kind of information will be encountered and how it will be expressed. There are so many ways in which definitions are conveyed in natural language that it is difficult to come up with a complete set of linguistic patterns to solve the problem of definition extraction. To make matters more complex, patterns are usually too broad, matching non-definitional contexts as well as definitional ones. For instance, the two sentences *The acid rain is a rain with significantly increased acidity as a result of atmospheric pollution* and *The acid rain is a problem with significantly increased consequences* could be both recognized as definitions, while only the first one can be considered a correct definition.

This Chapter reviews the state of the art in the area of automatic definition extraction. Before describing the systems focussing on definition extraction, the next Section will discuss the difference between definition and semantic relations extraction. In Section 3.3, systems based on specific knowledge domains or with a specific purpose are described. Particular attention will be given to systems developed in the area of e-learning, ontology building and Question Answering. Section 3.4 focus on a few systems having a broader approach to definition extraction, that are not limited to a specific area of knowledge and were developed using corpora covering different domains.

3.2 Semantic Relations and Definition Extraction

As we have seen in the previous Chapter, in recent years theoretical studies on definitions aim to identify different kinds of definitions based on the type of semantic relations conveyed by the sentences. Generally speaking, the automatic or semi-automatic extraction of defining information can be approached from two different points of view, based on two different main purposes:

- extraction of semantic relations (SRs)
- extraction of definitional contexts (DCs)

Some works have focused on the automatic extraction of SRs using dictionaries in both electronic and specialized texts, primarily as a support in the construction and organization of lexicons, terminologies, taxonomies and ontologies (Amsler, 1981; Nakamura & Nagao, 1988). Other studies have focused on the extraction of DCs for the development of resources such as dictionaries, lexical databases, terminological knowledge databases, or as a pre-process for the extraction of SRs (Pantel & Pennacchiotti, 2006). Moreover, in recent years the extraction of definitional knowledge has begun to influence the development of tools to improve the structure of the information present in the Web. Such is the case of the weight that has been given to the use of ontologies in the development of the Semantic Web (Auger & Barriére, 2008).

However, the methodologies for extracting SRs and DCs tend to agree in some respects, in such a way that the dividing lines between them can be blurred, mainly because of the similarity between their methodologies.

Therefore, and for the sake of establishing an overview of the state of the art in accordance with the current research, it seems to be necessary to make a brief mention of the specific characteristics distinguishing these two approaches to extraction of definitional knowledge.

Initial research in the area of automatic definition extraction, focusing on relation extraction, dates back to the late eighties Alshawi (1987). At that time, there was a considerable interest in the construction of lexical resources for NLP by exploiting Machine Readable Dictionaries (MRD). There is general agreement that lexicographical or terminological entries in MRDs can be used as a repository of lexical and taxonomic information that can be extracted from the respective definitions by using some kind of computational analysis of the text (Barnbrook, 2002).

One of the relations that has attracted more interest is the IS-A relation, that is the one clarifying the *genus* or the superordinate concept. Amsler (1981) was one of the first to take into account the usefulness of *genus* and specific difference in definitions. In his work, he presents a methodology for the elaboration of taxonomies from the identification of the definitions found in dictionary entries. Alshawi (1987) proposed a method mainly based on pattern searching to identify the *genus* in a definition, in order to extract and categorize entries using the Longman Dictionary of Contemporary English (LODCE). Pattern based systems were also used by Markowitz *et al.* (1986),

Jensen & Binot (1987) and Nakamura & Nagao (1988) in order to extract taxonomy relations by using different dictionaries.

The IS-A pattern has been used also to extract a set of semantic relations from definitions. For example, Vossen & Copestake (1993) used the pattern IS-A in order to obtain a set of semantic relations from analytical definitions. This study considered three relationships derived from this pattern: hyponymy-hypernymy, synonymy, and individuation. This works shows that the structure and function of MRDs make them well suited for pattern-recognition techniques. It is a matter of fact that MRDs were and are built following precise guidelines and do not present a large variability in the way their definitions are presented.

Going beyond MRDs, Hearst (1992) applied lexical-syntactic patterns to unrestricted domain-independent texts in order to extract taxonomic relations. In particular, he proposed a method for the automatic acquisition of hyponymy relations that are expressed in well-known ways (such as "NP as NP" (where NP stands for Noun Phrase), "NP and other NP", "NP especially NP", etc.). His study included a process to acquire new patterns by the extraction of the occurrences of two terms, where the semantic relationship bounding them was well known, by establishing a co-occurrence window between these two terms. All the contexts where the two terms occur are collected and then analyzed in order to build new patterns. This method was subsequently extended to cover other types of relations (Klavans & Muresan, 2000; Pearson, 1996; Prager *et al.*, 2001).

Other studies have used this same idea of using pairs of terms that share a semantic relationship in order to acquire new lexical-syntactic patterns. Morin (1999), for example, used this methodology for extracting hyponyms on a corpus of scientific texts in the area of agronomy.

Along with these works that exploited the idea of seeking occurrences of terms to find out new lexico-syntactic patterns, Pearson (1998) presented a study in which the possibility of using this methodology to acquire not only specific relations, but statements that fully describe the meaning of a concept is described. She proposed a simple extension of Hearst pattern $X = Y$ + distinguishing characteristics" where possible fillers for X were well formed terms (those word sequences following specific patterns), fillers for Y were terms or specific words from a particular word list (e.g.,

method, *technique*, etc.), and fillers for = were connective verbs such as *to be*, *consist* or *know*.

In the same line, Meyer (2001) noted that the occurrence of terms along with lexico-syntactic patterns can provide useful information to situate the term within a specific conceptual network and also to describe its attributes, thereby providing basic information about its meaning.

In general, the methodology to extract relations is based firstly on the identification of the type of relationships, then the next steps are 1) the discovery of patterns that explicitly express the intended relationship, 2) the search for the occurrences using the discovered patterns, and 3) the usage of semantic relations thus extracted as new instances in ontologies and terminology databases (Hearst, 1992; Pantel & Pennacchiotti, 2006).

Whereas, the methodology to extract definitions is based firstly on identification of an initial set of defining patterns, then the next steps are 1) the automatic extraction of definitional patterns, 2) the analysis of the results, in order to add other patterns to the initial paradigm and to include restrictions on patterns, 3) the application of the necessary changes, and 4) the repetition of these two last steps in order to improve the system (Klavans & Muresan, 2000; Sánchez & Márquez, 2005).

SR extraction is performed both over structured documents (MRDs, electronic encyclopedias) and unstructured documents (specialized texts), while extraction of DCs has been based mainly on unstructured documents. SRs extraction can be made using patterns as specific as required by the scope of the specific application at stake, while the DC extraction tends to be made with more general patterns that can be applied to a more encompassing area of knowledge. The expected result in the extraction of semantic relations is a defined and concise set of pieces of information between two terms or groups of terms, among which there is a specific relationship. Regarding definition extraction, the expected result is a set of pieces of information that also includes descriptions, conditions of use or pragmatic information that help to understand the meaning of the term.

In addition to this, the result of the extraction of semantic relations, structurally speaking, is a group of words or phrases, while in the case of the extraction of definitional contexts the result will tend to be a sentence or group of sentences.

Finally, the extraction of definitional contexts can be seen as an end in itself, or a step that can serve in the delimitation of semantic relations (Malaise *et al.*, 2004).

3.3 Specific Domain or Task

As previously referred, most of the systems aiming at automatic definition extraction focus only on a specific domain or a specific task, and for this reason they are hardly applicable to contexts other than the ones for which they were developed. In this section, we present an overview concerning the more interesting systems in the area.

With the objective of glossary construction for non-specialist in the medical area, Klavans & Muresan (2000) developed the DEFINDER system, based on a corpus composed by well-edited and structured full text consumer-oriented medical articles.[1] In this corpus, nearly 60% of the definitions were introduced by a limited set of simple text markers (such as punctuation marks), the other 40% being identified by complex linguistic phenomena (anaphora, apposition, conjoined definitions). The DEFINDER system is composed by two modules: 1) a pattern analysis module that performs shallow text processing using a finite state grammar, guided by cue-phrases (*is called*, *is the term used to describe*, *is defined as*, *is the term for*, etc.) and a limited set of text-markers; and 2) a grammar analysis module that uses a rich, dependency-oriented lexical grammar for analyzing more complex linguistic phenomena (e.g. apposition, anaphora).

The first module was dedicated to extracting definitions by searching for typographical and lexical patterns in conjunction with a finite-state grammar. This module also needed a filtering process, taking into account that the patterns could be a source of error, since it is used not only to handle definitions, but also to offer explanations and enumerations. The pattern analysis module was based on a part-of-speech tagger with a finite-state grammar for identifying medical terminology and for extracting definitions. The lexicon was augmented with the most frequent medical terms found in the corpus. This helped to eliminate incorrect tagging due to unknown words.

The second module used a combination of grammars and a statistical parser to find linguistic phenomena commonly used in the drafting of definitions in specialized texts. These phenomena include appositions, relative clauses or anaphora. The filtering

[1] From MEDLINEplus (http://www.cardio.com/articles.html).

module was added in order to remove some of the misleading patterns introduced by text markers (e.g. explanation, enumeration). The grammar analysis module was based on English Slot Grammar (ESG). The rich representation provided by ESG allowed for the identification of definitions introduced by more complex linguistic phenomena and not easily identifiable by shallow processing (Klavans & Muresan, 2001). At this point, there was the problem of differentiating a term and its definition when both elements were expressed, for example, by one noun phrase (a common structure in Synonymic Definitions). To solve this problem, the authors use a statistical method based on the frequencies of the candidate term and the definition, building on the hypothesis that the term is used a larger number of times in the text, while the definition is usually mentioned one time.

To analyze the results, the evaluation methodology is divided into three parts: assessing the performance of the system in terms of accuracy and coverage, evaluating the quality of the automatically generated dictionary judged by specialists and non-specialists, and comparing the results with on-line dictionaries.

Regarding the first evaluation, four non-experts marked definitions of a subsample of the corpus. Definitions marked by at least 3 of the 4 annotators were considered acceptable. DEFINDER identified 40 out of these 53 definitions, obtaining 0.87 precision and 0.75 recall. Besides the correct definitions (40), DEFINDER extracted 6 false positive definitions, thus decreasing precision. These results indicate that a large number of definitions automatically were recovered (75% of total) and they did not involve too much noise (only about 13% were no definitions).

Regarding the evaluation using experts, they used the on-line Medical Dictionary (OMD) as a comparative resource. Eight non specialists were asked to evaluate the definitions for 15 medical terms, where some definitions were extracted automatically and others came form the on-line dictionary. The purpose was to assign a quality rating in terms of usability (if the definition was useful in understanding the term) and clarity (corresponding to the degree of specialization of the definition) of the definition, on a scale of 1 to 7, where 1 was very bad and 7 excellent. Regarding usability, DEFINDER obtained a score of 5.17 while OMD obtained a score of 3.9. Regarding clarity, the score was 5,65 and 4.3 respectively. The same task was performed by a group of fifteen specialists obtaining similar results. These results show that DEFINDER was the preferred resource, since the subject in the evaluation experiment considered that the definitions

	UMLS	OMD	GPTMT
1	60% (56)	76% (71)	21.5% (20)
2	24% (22)	-	-
3	16% (15)	24%(22)	78.5% (73)

Table 3.1: Results in percentage of DEFINDER coverage against others dictionaries

extracted automatically by the system were easier to read and therefore more useful for understanding the meaning of the terms.

Finally, the definitions extracted by DEFINDER were compared with three other medical definition resources, namely the Unified Medical Language System (UMLS), Glossary of Popular and Technical Medical Terms (GPTMT) and the OMD. The purpose of this comparison was to highlight the fact that the developed system could serve as a complement to existing dictionaries. Ninety-three terms were automatically extracted by the system along with their definitions, and each definition was evaluated with respect to three different situations:

- The definition was the same in DEFINDER and in at least one of the other dictionaries.
- The definition was present in DEFINDER and in at least one of the other dictionaries, but it was different.
- The definition was not listed in any of the dictionaries.

When the definition extracted by DEFINDER is also found in one of the other dictionaries taken into consideration, these definitions are very similar, except for the definitions occurring in the UMLS; in this case the definition differs from the one extracted by DEFINDER. In other cases, the definition was very similar or it was not in the dictionary. Regarding the glossary GPTMT, almost 80% of the definitions found by DEFINDER had not been considered previously in this on-line resource.

In summary, the methodology of Klavans and Muresan is a system for the extraction of definitions from defining patterns on texts aimed at non-specialists. One of the most interesting contributions of this work is the evaluation methodology, which highlights the difficulty of defining a gold-standard when assessing what is considered as a good definition. It also highlights the importance of a system of this type for the improvement

and extension of lexical and terminological resources, supplementing these resources with data that has not been covered before.

The work of Klavans *et al.* (2003) is in some way related to the early research on machine readable dictionary parsing, but with a broader prospective. They used large on-line glossaries as corpus, focusing mainly on large government websites, with the aim of supporting Semantic Web and the linking between definitions from different sources. They developed different modules to deal with the various problems: one to identify glossaries in websites; another to extract definitions from free text (based on DEFINDER); a third one capable to parse definitions and load results in a relational database.

They used POS tagging and noun phrase chunking as source of the linguistic tools to annotate documents, and cue phrases to identify possible important information. In this task, several problems were identified, including the format of the definitions and the content in which they were presented.

In order to evaluate their system, a qualitative evaluation was carried out where human annotators were given 25 definitions and were asked to identify the *definiendum* and the defining phrase. The inter-annotator agreement was then calculated. Although there was a high agreement on the marking of the *definienda*, there was a lower agreement on the defining phrase, with 81% agreement if considering partial phrases (e.g. one annotator marks a full sentence, and another marks only part of it), and even lower if only exact matches were considered. This shows how humans differently perceive sentences to be definitions and the challenges of automatic definitions extraction.

In the context of e-Learning, but using the Web as a corpus, Liu *et al.* (2003) proposed a set of initial techniques for finding and compiling topic-specific knowledge (concepts and definitions) on the Web. The proposed techniques aimed at helping Web users to learn an unfamiliar topic systematically and in-depth. One of the ideas behind this approach is, by giving a concept, to retrieve those web pages containing the definitions of that particular concept and its sub-concepts. In order to identify definitions, a rule-based approach was used exploiting lexical and syntactic information as well as layout information (HTML tags).

The evaluation was based on the comparison between the result of their system and other two systems (Google and AskJeeves) on a list of 28 concepts. That is, for each concept and for each system, the list of retrieved pages was collected, then two

independent evaluators reported on the correctness of the first 10 retrieved definitions for each system. The overall precision for each system was 0.61 for the new system, 0.18 for Google, and 0.17 for AskJeeves.

Sánchez & Márquez (2005) developed a system for definition extraction in legal documents in Spanish. Their goal is to extract definitions by recurrent patterns and establish a database where the extracted definitions can be manipulated and made available through a user interface.

Initially, a linguistic analysis of legal texts was performed in order to determine what kind of patterns usually introduce definitions. This analysis resulted in the individuation of three groups for defining verbs, that were used to construct patterns manually. The system then identified the sentences where a definition pattern was present, and its constituents, that is definiens and definiendum, then it stored them in a database.

In order to evaluate the system, they extracted all the definitions introduced by a specific verb, that is *to mean*, ending up with 38 definitions, and then they calculated the precision and recall for the patterns dealing with this verb. In this way they obtained a precision of 0.97 and a recall of 1, but it is difficult to generalize the performance of this system for other patterns and for other domains.

Remaining in the domain of the law, but switching to Portuguese, Ferneda *et al.* (2012) using a corpus in the telecommunications regulations domain, developed a system based on machine learning algorithms, specifically with a Support Vector Machine classifier. In this case the system is not based on manually constructed patterns, but on features describing each sentence in the corpus. The set of features used to describe the sentences was determined by specialists. Examples of features are: (i) presence of verb *to be*, in any conjugation form, followed by an article, (ii) presence of the punctuation symbol *:* or the - and its position in the sentence, (iii) presence of the expressions *is defined as*, *is understood as* or *is called*. Sentences, where no feature was present were discarded; even so, the resulting dataset presented a high degree of imbalance, in fact, only 2% of the example were actual definitions. For this reason, negative examples were randomly discarded till matching the number of positive examples. With a balanced dataset, the system obtained a precision of 0.76 and a recall of 0.60, with an F-measure of 0.72.

3.3.1 LT4eL

Systems for semi-automatic glossary construction for an e-learning environment in several languages were built within the LT4eL project.[1] The aim of this European project was to develop multilingual language technology techniques and tools to be integrated into eLearning applications, to facilitate personalized access to knowledge within learning management systems and support decentralization and co-operation in content management (Monachesi *et al.*, 2006). In order to enhance the quality and speed of the learning process, different tools were developed, such as a keyword extractor, a domain ontology, a semantic search engine and a glossary candidate detector.

For each language involved in the project (namely Bulgarian, Czech, Dutch, English, Polish, Portuguese and Romanian), corpora were collected that were composed by learning materials focusing on Information Technology for non-experts. All the corpora were manually annotated and definitions were divided in three main categories: copula, verb and punctuation definitions. The first category includes all those definitions where the verb *to be* is used as connector verb. These are the most common definitions. The second group is formed by all those definitions in which other verbs are used as connectors, as for instance *to mean*, *to be called* but also *to be used to*. The third type are all those definitions introduced by specific punctuation marks, such as *:*, *-*.

For each language and for each category a rule-based system was developed, taking into consideration the syntactic and lexical structure of definitions marked in the corpora. In general, these grammars start with several simple rules which identify different parts of speech. Those rules can be combined in order to obtain more complex rules. After identifying the different structures, the specific rules for each definition type were created.

In Table 3.2 the performance of these grammars is indicated.

Regarding Slavic languages (Bulgarian, Czech and Polish), Przepiórkowski *et al.* (2007a) report that the best results were obtained for the Czech language, with a precision of 0.22 and a recall of 0.46. Authors explain the difficulty to obtain good results with a rule-based system for this kind of languages by referring the particular structure

[1] www.lt4el.eu

Language	Copula			Other Verbs			Punctuation			Total		
	P	R	F-m	P	R	F-m	P	R	F-m	P	R	F-m
Bulgarian	-	-	-	-	-	-	-	-	-	0.23	0.09	0.13
Czech	-	-	-	-	-	-	-	-	-	0.22	0.46	0.30
Dutch	0.21	0.92	0.34	0.26	0.41	0.32	0.77	0.03	0.05	-	-	-
English	0.17	0.58	0.26	0.34	0.32	0.33	0.33	0.12	0.17	-	-	-
Polish	-	-	-	-	-	-	-	-	-	0.23	0.32	0.27
Portuguese	0.32	0.66	0.43	0.14	0.65	0.23	0.28	0.47	0.35	0.23	0.75	0.35
Romanian	0.54	1.0	0.70	0.76	1.0	0.86	0.15	1.0	0.26	-	-	-

Table 3.2: Resuming LT4eL results

of these languages, as they are very rich in nominal inflection, with a large number of internal syncretism and relatively free word order.

The automatic definition extractors for Bulgarian, Czech and Romanian (Iftene *et al.*, 2008) received no further improved after the end of the project. As for the Portuguese extractor, it was the starting point for this doctoral research, and will be the object of the following Chapters.

Regarding Polish, in subsequent work the grammar was improved, obtaining a recall of 0.59 and precision of 0.19 and an F-score of 0.19 (Przepiórkowski *et al.*, 2007b). In order to improve the result for Polish, several experiments were carried out pairing the grammar with machine learning algorithms. In a first stage, several new grammars were developed where no rule to improve precision was inserted, in order to obtain a better recall score, ending with a grammar with recall of 0.88, precision of 0.11 and F-score of 0.28 (Degórski *et al.*, 2008b). In order to improve precision, machine learning algorithms were applied namely Naïve Bayes, C4.5, ID3, IB1, nu-SVC, and AdaBoost with Decision Stump (AB+DS). The data set was composed by all the sentences automatically annotated by the simple grammar. As attributes, the first 100 more frequent n-grams (n=1,2,3) composed by lemmas, syntactic categories and cases were selected. As the dataset was quite imbalanced, the distribution of classes was modified in order to obtain different ratios of 1:1, 1:5 and 1:10 by means of random under-sampling. The best result was obtained with AB+DS classifier with 0.18 of precision, 0.60 of recall and an F-measure score of 0.28. When the classifiers were replaced with ensembles of classifiers the best result was obtained using an ensemble composed by seven AB+DS classifiers, obtaining a precision of 0.20, 0.63 of recall and F-measure of 0.30. In both cases the dataset with the best performance was the one with a ratio of 1:1, that is a fully balanced dataset.

Finally it was opted to rely on machine learning only, resorting to Balanced Random Forest, (Degórski *et al.*, 2008a; Kobyliński & Przepiórkowski, 2008). This is a machine learning technique for classification using decision trees, where decisions are based on a subset of attributes which are randomly selected and the best attribute for the current tree is then chosen. Each tree is built using the same number of items from minority and majority class, thus coping with the issue of imbalanced datasets (Chen *et al.*, 2004). In the first experiment, this algorithm increased the F-measure score to 0.32 (with precision at 0.21 and recall at 0.69). In the second experiment, the algorithm was fine-tuned in order to improve the F_2-score, favoring recall over precision. In this way, an F_2-score of 0.43 was obtained, where the F_2-measure before the optimization step was 0.40.

Regarding the Dutch language, the same approach to improve the results obtained in the grammar was carried out (Westerhout & Monachesi, 2007). Differently form Polish, the focus was not on the learning algorithms but on the selection of features. Instead of only relying on n-grams for representing sentences, three groups of different features were selected: text properties (bag-of-words, bigrams, and bigram preceding the definition), syntactic properties (type of determiner within the defined term such as definite, indefinite or no determiner), proper nouns (presence of a proper noun in the defined term, e.g. location, person, organization, or no-class) (Westerhout & Monachesi, 2007, 2008). For copula definitions, using a Naïve Bayes classification algorithm and combining the different features, they obtained a recall of 0.67 and a precision of 0.80. In comparison with the syntactic pattern system developed previously, showing a recall of 0.92 and a precision of 0.21, the filtering phase brings an improvement in precision of 0.46 points, but a decline in the recall of 0.12. Looking at the corresponding F-measure scores, 0.34 and 0.73 respectively, it is clear that the machine learning filtering phase improved the overall performance of the system. Westerhout (2009) also applied Balanced Random Forest as a filtering module after the grammar, obtaining a precision of 0.77 and a recall value of 0.79, with an F-measure of 0.78. In this way, the automatic definition extractor for Dutch was further improved.

A very different approach to improve the grammar is the one proposed by Borg *et al.* (2009). They used genetic algorithms for weighting the manually-crafted linguistic patterns in order to obtain a fine-grained filter to select definitions. This resulted in a large improvement regarding precision from 0.17 (before the filtering stage) to 0.62.

The recall remained around 0.50 and the resulting F-measure value was 0.57. They also tried to automatically generate definitional patterns by means of genetic programming with a less encouraging result, obtaining a precision of 0.22, a recall of 0.39 and an F-measure of 0.28.

3.3.2 Ontology Building

Regarding the use of definition extraction with the aim of building or improving ontologies, there are several works worth mentioning (Malaise *et al.*, 2004; Storrer & Wellinghoff, 2006; Walter & Pinkal, 2006).

Malaise *et al.* (2004) developed a system for the extraction of analytical definitions containing hyperonym and synonym relations based on French corpora. The aim was to mine "defining expressions" in domain-specific corpora, and detect the semantic relations between their main terms in order to add structure to terminology. The extraction of definitions represents in this work the prior step for the development of differential ontologies, that are considered as standardized terminological hierarchical structures and represent the first step in the construction of any ontology. In this type of ontologies, each term is connected with its definition, and it is associated with differential characteristics. These characteristics consist of: a) the similarity with relatives (semantic features that it shares with its hypernym term), b) the difference from their relatives (the characteristics that distinguish the term of others who share the same hyperonym); c) the similarity with the sibling (the characteristics it shares with its co-hyponyms), d) the difference with the sibling (the characteristics that differentiate it from other co-hyponyms). They used a training corpus with documents from the domain of anthropology and a test corpus from the domain of dietetics.

The methodology of this study was based primarily on the extraction of non-specialized definitions in order to gather information relevant to the development of an ontology, specifically on the vertical axis (hyponyms), the horizontal axis (co-hyponyms) and the transverse axis (relationships across the same domain).

Differently from Hearst (1992), these authors first analyzed the definitions marked in the corpus and, based on this, designed and tuned the lexico-syntactic patterns. These patterns were based on the information output by the parser, that is lemma, morpho-syntactic category and grammatical function. They used lexico-syntactic markers and patterns to detect at the same time both a definition and its main semantic relation.

In this way, each pattern accomplished three different tasks: extraction of the defining sentences, selection of a semantic relation and extraction of the inter-defined terms. The patterns focused on definitory verbs such as *to denote*, *to define*, lexical markers such as *definition*, *term*, discourse markers such as *that is*, and punctuation marks, such as parentheses.

The patterns were applied to the corpus in order to identify terms that were defined and related definitions. This identification of the constituent parts of the definition was done along two criteria: taking into account the position they could occupy according to the defining pattern; and taking into account the morphosyntactic category of the words composing the pattern. Finally, the semantic relationship type was automatically identified. For this purpose, each defining pattern was linked to a particular relation. The relationship could be of four types: language (synonyms and antonyms), hierarchical (hypernyms), transverse (meronymy or causal relationships), and horizontal (relations representing two or more terms in the same level).

The evaluation of the system using a corpus of a different domain makes the results more interesting as this puts the system under a more stressing evaluation. Nevertheless, it is not clear what is the nature and purpose of the documents making this corpora, namely if they are consumer-oriented, technical, or scientific papers, etc. This approach was evaluated in two parts: automatic extraction of definitions, and automatic identification of conceptual relationship. As for the automatic extraction of definitions, they report a precision of 0.55 and a recall of 0.39.

In this case, the authors note that the best results were obtained by those definitions matching more than one defining pattern. The quality was determined by the number of patterns present in the context: the more patterns were holding, the greater the probability that the context was a definition. When considering the correct identification of semantic relations, for the two different relations, hyponym and synonym, they obtained, respectively, 0.04 and 0.36 of recall, and 0.61 and 0.66 of precision.

In short, the work of Malaise *et al.* (2004) introduces a methodology for the extraction of definitions as a step towards building differential ontologies. Thus, their interest is directed not only to the descriptions about the meaning of a term that can be extracted by the automatic search for definitions, but also to specific semantic relationships that can occur between the term being defined and the terms present in the definition.

Walter & Pinkal (2006) used a corpus composed by court decisions, restricted to environmental law. Their aim was to improve the quality of a text-based ontology learning application. Based on the analysis of their corpus, they identified some common structural elements which constitute a definition. Besides the defined term, the definitory expression and the connector, in this particular domain, a definition is also characterized by a term delimiting the domain field and a distinctive feature that in German is the word *dann*, with no equivalent in English or Portuguese. Bearing in mind the broad linguistic variations of definitions, they manually built 33 rules plus a set of filtering rules.

In order to evaluate the system, two domain experts judged the candidate sentences correctness, obtaining a precision of 0.45 and 0.49, respectively. Precision improved considerably (over 0.70) when the best 17 rules were used out of the 33. As the corpus was not annotated, recall was not calculated.

Storrer & Wellinghoff (2006) reported on an approach to automatically detect and annotate definitions for technical terms in German corpora, with the aim at automatically extracting semantic relations between the *definiendum* and the *definiens*. The main feature they investigated was the connector verb or, as they called it, the definitory verb. The corpus consisted of 20 technical papers in the field of information technology with a total of 103.805 words. They first annotated the corpus manually, focusing on Aristotelian definitions and identifying the *definiens*, *the definiendum* and the connector verb. As a result, they marked 174 definitions in the corpus, which were used as the gold-standard and empirical basis of the study on the feasibility of extraction of semantic relations from definitions.

Differently from other works, they used valency frames in order to structure their rules. Valency frames are composed by linguistic information stating which arguments a verb takes, such as Subject, Object, in which position and with which prepositions. This is a rule-based expert-driven approach, with all information being provided by human experts (valency frame, categorization of definitions).

Definitions in technical texts tend to be well-structured, frequently matching crisp rules therefore making such an approach viable. Classifying definitions into categories according to their valency frame allows to concentrate on specific patterns per rule, taking a fine-grained approach rather than a generic definition extractor.

In this way, they defined a paradigm of 19 patterns having at this core verb forms such as, *know as*, *call*, *defined as*, *known as*. After the extraction of the definitions, the system identified semantic relations using a different set of patterns.

The evaluation was carried out using the gold-standard, and achieved an average of 0.34 precision and 0.70 recall, with a big variation among different patterns. Coverage was significantly higher when the defined verb in the pattern used was followed by a preposition. Some patterns are more problematic, as they can be used in a variety of contexts that do not provide defining information.

3.3.3 Question Answering

If we focus on the Question Answering (QA) field, several studies were undertaken in order to answer definition questions. Answering definition questions is a challenge for question answering systems. Differently from most of the work in QA focusing on factoid questions, where strong predictions about the type of expected answer (i.e. a date, a name of a person, an amount, etc.) are possible, definition questions require a different approach, as a definition can be a phrase or a sentence for which only very global characteristics hold. At the same time, it is worth noting that the main difference between this task and our doctoral research work resides in the fact that we do not know beforehand the *definiendum*. This lack of information makes the task more difficult because it is not possible to use the term as a clue for extracting its definitions.

Most of the works presented in this section were carried out using the data made available by the Text Retrieval Conference (TREC) workshop series. TREC is an annual conference that encourages research in information retrieval and related applications by providing a large test collection, uniform scoring procedures, and a forum for organizations interested in comparing their results. When the definition extraction task is carried out in the context of QA, the *definiendum* is already known, or it can be deduced from the question. This can turn the overall task of definition extraction easier.

Prager *et al.* (2001) report on the use of a very simple heuristic. Using WordNet as an external resource, they looked for those hypernyms which co-occur with the definition term in a snippet. Co-occurring terms with the highest frequency and closest relation in Wordnet were ranked highest. They argue that higher ranked co-occurring terms are more likely to be considered acceptable hypernyms and can be presented as a definitional answer. Using this heuristic they were able to answer correctly to a set of questions

based on TREC9[1] 82% of the times. The problem in this approach is that not all definitional questions can be answered with a hypernym.

Joho *et al.* (2001) extended the work done by Prager *et al.* (2001) by developing patterns able to extract definitional information or hypernyms not present in Wordnet. Their method extracted all the sentences where the query noun was found and then ranked the sentences according to (i) the sentence position (the earlier it occurs in the document, the more likely it is to be a definition); (ii) word count of the highest occurring words in all candidate answers (excluding stop words); and (iii) a weight of the pattern being annotated. The system was found to be highly effective at locating good definitory context, finding at least one high quality definition in the top 10 returned sentences for 82% of test queries.

Saggion (2004) combined probabilistic techniques with shallow linguistic analysis and tested his method in the TREC QA 2003,[2] focusing on definition questions.

In a first stage, 20 candidate texts were collected by means of a probabilistic document retrieval system that took the definition question as input. In order to obtain a higher recall, the question was expanded with a list of related terms using WordNet and the Encyclopedia Britannica. The process of acquiring occurrences of the search term in lexicographical resources on the web was intended to acquire secondary terms that usually appear in the definitions of the search term. In a second stage, the candidate sentences were analyzed in order to match a list of 50 definition patterns created manually.

Regarding the results obtained at the TREC QA 2003 competition, this autor reported an F_5 -measure, where the recall is 5 times more important than precision. His system obtained an F_5-measure of 0.24, where the best score in the same competition was of 0.56 and the median was of 0.19. Significantly, this work represents an effort to integrate different lexical resources in order to improve definition extraction.

Tjong *et al.* (2005) focused their work on a specific domain field, that is medical texts about Repetitive Strain Injury (RSI) in the Dutch language. They exploited two different strategies based on syntactic and layout information. As the domain was very specific, it was possible to identify a set of question types in correspondence with expected answers such as treatment, symptom, definition, cause, diagnosis and

[1] http://trec.nist.gov/data/qa/t9_qadata.html
[2] http://trec.nist.gov/data/qa/t2003_qadata.html

prevention. The syntactic parser developed for this system was applied to the first sentence of the articles of a medical encyclopedia.

Definitional answers were extracted on the basis of the concept being present as the *definiendum* followed by the verb *to be*, and achieved a precision of 0.18. Symptom answers were extracted on the basis of the presence of particular phrases (*a sign of*, *an indication of*, *is recognizable*, *points to*, *manifests itself in*, etc.), and achieved a precision of 0.76. Cause answers extracted on the basis of the presence of certain phrases (*causes*, *cause of*, *results of*, *arises*, *leads to*), achieved a precision of 0.78. The extracted sentences were then made available to a QA system and if no answer was found within that set, it used a generic QA system as a fallback option.

More recently machine learning techniques have been combined with pattern recognition in order to improve definition extractors.

Miliaraki & Androutsopoulos (2004) used a machine learning-based method to identify 250-character single-snippet answers to definition questions using a collection of documents. Their method combined and extended two different techniques based mostly on manually crafted lexical patterns and WordNet hypernyms.

In a first experiment, they applied the Support Vector Machine algorithm (SVM), using as features those attributes highlighted by Joho & Sanderson (2000), such as the sentence position, word count, lemmas and manually crafted patterns. The second experiment included additional features, taking into account the existence of hypernyms, as in the work of Prager *et al.* (2001). The third and final experiment included n-grams after or before the defined terms.

In order to evaluate the system, the TREC-9[1] and TREC-2001[2] data were used. The best performance was obtained using the third experimental setting, with a precision of 0.73 and 0.85, respectively, for the two corpora.

Similarly, Blair-Goldensohn *et al.* (2004) integrated pattern matching and machine learning techniques in order to answer definition questions from generic corpora. They grouped definitions in seven categories:

- Genus: a generic description of the category the term belongs to;
- Species: specific information which distinguishes the term from other terms in the same category;

[1] http://trec.nist.gov/pubs/trec9/t9_sysdes.html
[2] http://trec.nist.gov/data/qa/t2001_qadata.html

- Target partition: dividing the term into sub-parts;
- Cause: the cause of something;
- History: historical information about the term;
- Etymology: information on the term's origin;
- Non-specific definitional: any type of information related to the term (this can be seen as a superset of all the above categories, and more).

Based on this classification, they manually collected and annotated a training corpus composed by documents retrieved from the web by using a list of 15 definition terms. In a first experiment, features such as term concentration and position within a document were used and 0.81 accuracy was achieved. In the second experiment, 18 patterns were used, leading to a precision of 0.96.

Fahmi & Bouma (2006) worked on a Dutch medical QA system. The aim of this work was to investigate to what extent machine learning techniques could be used to distinguish definitions from non-definitions in a corpus of sentences containing a subject, copular verb, and predicative phrase. They experimented with different algorithms (namely Naive Bayes (NB), maximum entropy (ME), and three Support Vector Machines (SVM)). They propose several attributes to classify definition sentences, namely text properties (such as n-gram and bag-of-words), sentence position, syntactic properties (position of the subject of the sentence and whether the subject contains a determiner) and named entity classes. By combining these features, they obtained eleven different configurations.

In order to train the classifiers, a corpus derived from medical pages of the Dutch Wikipedia was used. They extracted sentences on the base of simple syntactic patterns, ending up with 2,299 sentences, of which 1,366 were actual definitions, thus obtaining an initial accuracy of 0.59. Running the experiment with the eleven configurations, their baseline was improved to 0.92, with maximum entropy, using a features configuration consisting of both syntactic properties and sentence position.

Chang & Zheng (2007) built a system to extract definitions using off-line documents. These authors manually collected and labeled the definitions in documents containing the definitions of computer science terms. The final corpus was composed by 5566 sentences, of which 2566 were definitions, where 255 terms were defined. These sentences were taken as training data to train the definition extraction model. As features to

represent definitions they used 1) the number of words contained in the definition; 2) the position of the definition in the document; 3) the number of terms included in the definition; 4) the presence of specific words or sequence of words, based on a list of words previously built on the base of the training data; 5) a combination of words and POS. They experimented with three different algorithms, namely Naïve Bayes, Decision Tree and Support Vector Machine (SVM), obtaining the best score with SVM with an F-measure of 0.83 (precision of 0.94 and recall of 0.75)

3.4 General Approaches

Focusing specifically on the extraction of definitional contexts, the majority of the systems that automatically extract definitions have been built taking into account corpora on a specific knowledge field and mostly for a specific application. Few works have used big corpora covering different knowledge domains, without a specific user context.

Rebeyrolle (2000) and Rebeyrolle & Tanguy (2000) reported on work for the French language, where they used a big corpus (more than 900,000 words) composed by a geomorphology handbook, scientific articles on knowledge engineering, internal documents developed by companies and a collection of articles from the *Encyclopédia Universalis*. After analyzing the definitions presented in these specialized texts, these authors proposed a typology of definitional contexts (DCs) along with a formal linguistic representation, describing a methodology for their extraction from morpho-syntactic patterns. They tried to extract DCs automatically from taking into account three different verbs or verbal structures: patterns containing the verb *to define*, patterns containing the verb *to signify* and its synonymous, and patterns containing the verb *to be*, ending up with 8 different patterns.

In order to evaluate their system, they manually marked all the definitions in the corpus. In this way they could calculate precision (P) and recall (R) for the definitions automatically extracted using the patterns. Table 3.3 shows the results for each pattern.

Patterns 1, 2, 4, 5 and 6 obtained the best recall score, this means that all the definitions characterized by these patterns, present in the corpus, were automatically extracted. Regarding precision, patterns 3 and 4 got the highest value, which means that most of the sentences recovered here were effective definitions. By contrast, patterns 1

Defining Patterns	P	R
1 - défini$	0.051	1
2 - définir * comme	0.733	1
3 - définir & comme	0.926	0.909
4 - définir (Non Vbe) * comme	0.917	1
5 - (signifier | vouloir | dire | entendre)	0.419	1
6 - (signifier | vouloir | dire | entendre) * par	0.486	1
7 - (est|sont)(un|une|le|la|les|l'|des)	0.168	0.935
8 - être * (det | num)	0.039	0.997

Table 3.3: Results for each definitional pattern presented in the work of Rebeyrolle & Tanguy (2000)

and 8 have a much higher degree of noise, since over 84% of all candidates automatically retrieved are not definitions.

Authors report that if a pattern is broaden with the inclusion of a window looking to the next adverb, there is a significant increasing in accuracy, even if the impact on precision and recall resulting from different patterns can differ. In particular, the use of a window of 6 words produces an improvement in precision value but it lowers recall. In the case of pattern 7, composed by the verb *to be*, the restriction on the tense and grammatical person, like the inclusion of a window till the next article, increases precision, although both precision and recall remain low compared with those obtained by other definitional patterns.

In short, the methodology used aims to formalize patterns with certain types of verbs or verbal structures, and extracts these patterns on a corpus of specialized texts. Results demonstrate that some patterns are more efficient than others, which are likely to return a lot of noise. Even if this study uses a huge corpus covering different domains, it appears very limited in terms of the variety of definitions it deals with, as it focuses on a very restricted type of verbs, whose use is considerably restricted to definitory sentences. When a less restricted verb is used, such as *to be* results are quite discouraging.

Another study dealing with analytical definitions is the one carried out by Acosta *et al.* (2011), whose focus is on definitions introduced by the verb *to be* and other similar verbs such as *to characterize*, *to consider*, *to define*, etc. Using a corpus made of medical text in Spanish, the authors built a chunk grammar using these verbs together with lexical information, such as the presence of lexical expressions indicating a hierarchical

Verbal Patterns		P	R
Concebir (como)	To conceive (as)	0.67	0.98
Definir (como)	To define (as)	0.84	0.99
Entender (como)	To understand (as)	0.34	0.94
Identificar (como)	To identify (as)	0.31	0.9
Consistir de	To consist of	0.62	1
Consistir en	To consist in	0.6	1
Constar de	To comprise	0.94	0.99
Denominar también	Also denominated	1	0.87
Llamar también	Also called	0.9	1
Servir para	To serve for	0.55	1
Significar	To signify	0.29	0.98
Usar como	To use as	0.41	0.95
Usar para	o use for	0.67	1
Utilizar como	To utilize as	0.45	0.92
Utilizar para	To utilize for	0.53	1

Table 3.4: Results for each definitional pattern presented in the work of Alarcón *et al.* (2009)

semantic relation (i.e. *type*, *kind*, *subtype*, *class*, *example*). They did not show results for each verbal pattern, but only the final performance of the system, which obtained a precision of 0.57 and a recall of 0.68, with an F-measure of 0.62.

A similar approach, but with a broader coverage of patterns, including also functional and extensional definitions, is offered by Alarcón *et al.* (2009), based on a Spanish corpus composed of over 1,000 documents covering eight different domains, namely Law, Human Genome, Economy, Environment, Medicine, Informatics and General Language. The authors manually developed a set of syntactic and verbal patterns to support the automatic extraction of definitions. The starting point of this process was a list of 163 terms extracted from a dictionary on the human genome. Then all the sentences in the Human Genome sub-corpus were extracted and analyzed, looking for definitory patterns. They end up with a list of 29 definitory verbs, and each verb was paired with a set of restrictions regarding its realization in a definitory sentence, such as tense, person, following adverbs.

Using these patterns, these authors built a system composed by three modules. The first module automatically extracts the sentences by resorting to the definitional patterns manually constructed. The second module filters the output of the first one by

applying a rule-based system. Finally, there is a third module that marks the *definiens* and the *definiendum*. The performance of the system was calculated for each different pattern. Table 3.4 shows the results for some of the most significant patterns. As with Rebeyrolle & Tanguy (2000) there are big differences among patterns. In terms of recall the performance is quite good, with values ranging from 0.87 to 1. In terms of precision there is more variability, the best score being 0.94, while the worst 0.29.

3.4.1 From Terms to Definitions

We have seen that in QA, the term whose definition is to be found is known beforehand. This characteristic is not exclusive of this application, but it is possible to find other systems were the *definiendum* is already given. Here, two such systems are described, where the definition extractor tool follows a term extractor module.

Turning more specifically to the Portuguese language, it is worth mentioning a tool not directly aimed at the extraction of definitions, but that offers this functionality. This tool is called CORPÓGRAFO (Pinto & Oliveira, 2004) and has been implemented as a web environment that assists, in various aspects, the work with corpora, such as compilation and terminology extraction, or the development of ontologies. One of the functionalities of this tool is the semi-automatic extraction of definitions. The semi-automatic definition extraction module follows a terminology extraction module. That is, once the user enters the documents he wants to work on, the module for detecting terms gives back a group of term candidates that the user can validate manually. Once this group is validated, each term is combined with a number of defining patterns in order to search for definitions in the text. These patterns were developed manually by observing all the sentences where terms, returned by the term extractor, appear, using a corpus on Neuroscience.

The system was tested on a corpus in the field of Medicine. Terms were extracted and followed by the respective definitions. Authors present results for each term. For example, the term *fibromyalgia* has 31 definitions in the evaluation corpus. Out of this number, the system proposes 33, of which 29 cases were correct definitions. The list of compiled patterns is very basic and it does not account for the variety of formulas that can be used to express a definition. However, this basic list can extract a first set of candidate definitions on which the user performs a manual validation to verify the utility of each outcome.

The other system worth addressing in the present section is GlossExtractor. This is a web application developed for English, whose function is to extract a list of candidate definitions from various types of documents on the Internet (on-line glossaries, web documents or pages specified by a user), resorting to a list of terms previously extracted (Navigli & Velardi, 2007). This system allows the user to define the search scope and the terms on which the definition extractor works.

GlossExtractor is composed of three modules: 1) extraction of candidate definitions, 2) identification of the best candidates based on stylistic filters and 3) a rule-based filter. The system starts from a list of terms that is entered by the user or that resulted from a previous terminology extraction process. This list is searched first in on-line glossaries previously indexed. Then, each term in the list is paired with a list of very simple definitory patterns, such as "*term* is a", "*term* define","*term* refers to", "*term* is a kind of". Each sequence forms a search expression launched into Google, and for each one, the first five returned pages are considered and are converted to plain text. After obtaining these documents, every sentence in the documents matching the simple patterns is considered a candidate definition, resulting in a significant increase of noise. The next two steps try to eliminate the noise. The first is a filter based on patterns, matching analytical definitions, intended to identify the *genus* and the distinctive characteristics. It is worth to note that unlikely all other system, the regular expressions of this filtering module were not created manually but they were automatically learned using a decision tree machine learning algorithm, namely the J48 algorithm from the Weka (Witten & Frank, 2005). As input features for the algorithm, they used a representation based on the 5-gram after the defined term.

The last step is a filter that removes definitions that are not relevant to the domain of interest. This is accomplished by creating a probabilistic model based on the set of words forming the terminology, and assigning a probability of occurrence to each word. As a result, GlossExtractor returns a list of candidate definitions, which the user can validate, deleting the incorrect ones.

The evaluation of the system was carried out, firstly, to assess the effectiveness of stylistic and domain filters by using a gold-standard, and secondly, to assess the process of automatic production of a glossary from a web-site. In order to create a gold-standard, definitions were extracted from professional glossaries. Starting from the terms defined in these glossaries, new definitions were automatically extracted from

	Precision	Recall	F-measure
Glossaries	0.90	0.89	0.89
Web Documents	0.92	0.81	0.86

Table 3.5: Results of GlossExtractor presented in the work of Navigli & Velardi (2007)

the Web using GlossExtractor. Both groups of definitions were manually evaluated in order to separate good and bad candidates.

The results presented in Table 3.5 are for the training corpus. In general, higher rates can be observed for both precision and recall for both experiments.

3.5 Conclusions

One of the common characteristics in the methodology for automatic definition extraction is the search for definition patterns, emerging from the analysis of the occurrences of definitions in specialized texts. From these studies, it was observed that, in contexts where information is provided about a term, a number of syntactic and typographic structures are used that could serve as a starting point for the automatic extraction. These structures can be grouped, generally, in typesetting and syntactic patterns.

The typesetting patterns can be the punctuation marks, as they often function as connectors between the term and its definition, and the typesetting of the text, which is used to highlight important information. On the other hand, syntactic patterns can consist of verbs introducing a definition, of discourse markers or a mix of both, thus forming more complex patterns.

Regarding syntactic patterns, almost all studies make use of verbal patterns composed by verbs such as *to mean*, *to define*, *to signify*, etc, with a different degree of complexity, introducing in the pattern a window of words after the verb, or explicitly stating the adverbs after the verbs, its tenses, etc. Some works take into consideration also verbal patterns indicating different semantic relations (e.g. causal, functional, etc.). Few studies use other lexical patterns besides verbal ones, such as *is the term for*, *type of*, *kind of*.

When results are presented for each different pattern, it is clear that there are some patterns that present more noise than others. This is because some verbal constructions, such as *is defined as*, are used mostly in a definitional context, while others, such as *is*

a, are quite common in all kind of discourses and tend to produce a greater amount of false positives. For this reason, some systems resort to a filtering stage, either by using more complex patterns or machine learning algorithms.

To summarize, there is a lack of general heuristics to use as guideline in the development of a definitions extractor. This means that, if there is a need of a new extractor, the process must start almost from scratch, starting to analyze a possible set of definitions and building specific patterns.

Chapter 4

Punctuation Definitions: Rule-Based Heuristics

4.1 Introduction

This Chapter focus on punctuation definitions and the rule-based approach. The development phase is carried out using a corpus written in Portuguese, while the testing phase uses both a sub-corpus in Portuguese and a corpus in English.

This approach will be also applied to other types of definitions, that is copula definitions and verb definitions, in order to determine a starting point or baseline against which to compare results obtained with more advanced techniques, presented in the next Chapters.

The research work described in this Chapter aims to develop a methodology for automatic definition extraction composed of different heuristics addressing different types of definitions. The distinctive characteristics of this rule-based approach adopted here is the focus on the part-of-speech information rather than on the lexical information. If pattern-based grammars are implemented using mostly lexical information, that is, the actual words occurring in definitions, their performance deteriorates when they are applied to contexts different from the development one.

This Chapter starts with the presentation of the corpora used in the development of heuristic and method for automatic definition extraction and with the description of the annotated definitions. Then the rule-based method is described, first its general characteristics and then the specific aspect when applied to the punctuation definitions. Results for this type of definitions are presented for both Portuguese and English lan-

guages. Finally the same heuristics are applied to copula and verbal definitions. Results for these two types of definitions are used as baselines for the work described in next Chapters.

4.2 The Corpora

As mentioned above in the Chapter about the state of the art, the starting point in the development and testing of effective heuristics for automatic definition extraction is to be found in a corpus with manually annotated definitions. In this research, two corpora from different languages, namely Portuguese and English, are used. A third corpus, from Dutch, is used in the next Chapter, to test the copula definition machine learning module, where it will be described in detail.

It is worth referring that, though written in different languages, these corpora are comparable as they were collected and annotated following the same guidelines. They were built in the context of the Language Technology for e-Learning (LT4eL) project[1] funded by European Commission, with the purpose of enhancing Learning Management Systems (LMS) with new functionalities, including an automatic definitions extractor.

The documents in the corpora include learning materials in information technology written by experts for beginners or for relative experts. For our purpose, they are easily usable and compareble given that they are annotated with the same type of morphosyntactic information across different languages. Given that all the documents were originally in several different file formats (.pdf, .html, etc.), they were pre-processed in order to obtain a uniform corpus annotated with linguistic information. In a first phase, the original documents were converted into a common XML format, conforming to a DTD derived from the XCES DTD for linguistically annotated corpora (Ide & Suderman, 2002). In this format, structural and layout information is retained as XML tags.

The main corpus, used for developing and testing is the one written in Portuguese, composed of 33 documents with a total of about 270,000 tokens. The documents are tutorials, masters and doctoral dissertations, and research papers in the general domain of Information Technology. Within this general domain, it is possible to individuate three sub-domains: Information Technology for non experts, e-Learning, and Information

[1] http://www.lt4el.eu/

Domain	Tokens
Information Society	92,825
Information Technology	90,688
e-Learning	91,225
Total	**274,738**

Table 4.1: Portuguese corpus composition in three different sub-domains.

Society. One third of the corpus is composed of documents that are mainly tutorials focusing on basic notions and tools in the domain of computer technology (tutorials about the use of text editors, HTML, Internet, etc.). Another third of the corpus is composed of documents (mainly articles and thesis) on e-learning concepts (practices and governmental policies). The last third is the Section 3 of the Calimera guidelines. These guidelines have been compiled by the CALIMERA[1] Co-ordination Action, funded by the European Commission, with the goal of explaining, in an accessible way, how technologies can be deployed to support digital services designed to meet real user needs. Table 4.1 summarizes the composition of the corpus.

To obtain part-of-speech (POS) tags, lemmas, and morpho-syntactic information, the corpus was automatically annotated using the LX-Suite (Silva, 2007), a set of tools for the shallow processing of Portuguese with state-of-the-art performance. This pipeline of modules comprises several tools, namely a sentence chunker (99.94% F-score), a tokenizer (99.72%), a POS tagger (98.52%), and nominal and verbal featurizers (99.18%) and lemmatizers (98.73%).

In Figure 4.1, a sample of the final corpus format is presented. Each sentence is delimited by the tag `s`, and each token by the tag `tok`. Of particular interest for the development of rule-based grammars are the attribute `base`, containing the lemma of each word, the attribute `ctag`, containing the POS information, and the `msd` with the morpho-syntactic information on inflection.

The English corpus is a collection of 7 tutorials with a total size of 287,910 tokens and 20,172 sentences. The corpus is annotated with linguistic information, using the Stanford POS tagger (Toutanova & Manning, 2000). The annotation process and the final format are the same as the one for the Portuguese language, the only difference being in the tag-set used by the linguistic annotation, that is the tags used to mark the

[1] http://www.calimera.org

```
<par id="p197" rend="P class=MsoNormal style=text-align:justify">
  <s id="s329">
    <tok base="ftp" class="word" ctag="PNM" id="t10354" sp="y">FTP</tok>
    <tok base="ser" class="word" ctag="V" id="t10355" msd="pi-3s" sp="y">é</tok>
    <tok base="um" class="word" ctag="UM" id="t10356" msd="ms" sp="y">um</tok>
    <tok base="protocolo" class="word" ctag="CN" id="t10357" msd="ms" sp="y">protocolo</tok>
    <tok base="que" class="word" ctag="CJ" id="t10358" sp="y">que</tok>
    <tok base="possibilitar" class="word" ctag="V" id="t10359" msd="pi-3s" sp="y">possibilita</tok>
    <tok base="a" class="word" ctag="DA" id="t10360" msd="fs" sp="y">a</tok>
    <tok base="transferência" class="word" ctag="CN" id="t10361" msd="fs" sp="y">transferência</tok>
    <tok base="de" class="word" ctag="PREP" id="t10362" sp="y">de</tok>
    <tok base="arquivo" class="word" ctag="CN" id="t10363" msd="mp" sp="y">arquivos</tok>
    <tok base="de" class="word" ctag="PREP" id="t10364" sp="y">de</tok>
    <tok base="um" class="word" ctag="UM" id="t10365" msd="ms" sp="y">um</tok>
    <tok base="local" class="word" ctag="CN" id="t10366" msd="ms" sp="y">local</tok>
    <tok base="para" class="word" ctag="PREP" id="t10367" sp="y">para</tok>
    <tok base="outro" class="word" ctag="ADJ" id="t10368" msd="ms" sp="y">outro</tok>
    <tok base="por_" class="word" ctag="PREP" id="t10369">por_</tok>
    <tok base="a" class="word" ctag="DA" id="t10370" msd="fs" sp="y">a</tok>
    <tok base="internet" class="word" ctag="CN" id="t10371" msd="fs">Internet</tok>
    <tok class="punctuation" ctag="PNT" id="t10372" sp="y">.</tok>
</s>
</par>
```

Figure 4.1: The sentence "*FTP é um protocolo que possibilita a transferência de arquivos de um local para outro pela Internet (FTP is a protocol that allows the transfer of files from one location to another on the Internet).*" in XML format

part of speech and morpho-syntactic information contained respectively in the attributes *ctag* and *msd*. Figure 4.2 shows an example of a definition in the English corpus.

4.2.1 Annotated Definitions

The definitions in the corpora are manually annotated by human annotators. The definitions taken into account in the present work are of three types:

Punctuation definitions (`punct_def`): introduced by punctuation marks, as in: *TCP/IP: protocolos utilizados na troca de informações entre computadores (TCP/IP: protocols used in the transfer of information between computers).*

Copula definitions (`is_def`): where the connector verb is the verb *to be*, as the example: *FTP é um protocolo que possibilita a transfêrencia de arquivos de um local para outro pela Internet (FTP is a protocol that allows the transfer of archives from a place to another through the Internet).*

Verbal definitions (`verb_def`): introduced by a verb other than *to be*, as in: *Uma ontologia pode ser descrita como uma definição formal de objectos (An ontology can be*

```
<s id="s3791">
<tok base="simulation" class="word" ctag="NN" id="t75325" msd="N,SG" sp="y">Simulation</tok>
<tok base="program" class="word" ctag="NN" id="t75326" msd="N,SG,proper,vrbl" sp="y">program</tok>
<tok base=":" class="other" ctag=":" id="t75327" msd="" sp="y">:</tok>
<tok base="a" class="word" ctag="DT" id="t75328" msd="DT,SG,wh" sp="y">A</tok>
<tok base="computer" class="word" ctag="NN" id="t75329" msd="N,SG" sp="y">computer</tok>
<tok base="program" class="word" ctag="NN" id="t75330" msd="N,SG,proper,vrbl" sp="y">program</tok>
<tok base="that" class="word" ctag="WDT" id="t75331" msd="CJ" sp="y">that</tok>
<tok base="simulate" class="word" ctag="VBZ" id="t75332" msd="V,PRES,S,finite" sp="y">simulates</tok>
<tok base="an" class="word" ctag="DT" id="t75333" msd="DT,SG,wh" sp="y">an</tok>
<tok base="authentic" class="word" ctag="JJ" id="t75334" msd="AJ,ABS,vrbl" sp="y">authentic</tok>
<tok base="system" class="word" ctag="NN" id="t75335" msd="N,SG,proper,vrbl" sp="y">system</tok>
<tok base="(" class="punc" ctag="(" id="t75336" msd="" sp="y">(</tok>
<tok base="city" class="word" ctag="NN" id="t75337" msd="N,SG,proper,vrbl" sp="y">city</tok>
<tok base="," class="punc" ctag="," id="t75338" msd="" sp="y">,</tok>
<tok base="pond" class="word" ctag="NN" id="t75339" msd="N,SG,proper,vrbl" sp="y">pond</tok>
<tok base="," class="punc" ctag="," id="t75340" msd="" sp="y">,</tok>
<tok base="company" class="word" ctag="NN" id="t75341" msd="N,SG,proper,vrbl" sp="y">company</tok>
<tok base="," class="punc" ctag="," id="t75342" msd="" sp="y">,</tok>
<tok base="organism" class="word" ctag="NN" id="t75343" msd="N,SG" sp="y">organism</tok>
<tok base=")" class="punc" ctag=")" id="t75344" msd="" sp="y">)</tok>
<tok base="and" class="word" ctag="CC" id="t75345" msd="CJ" sp="y">and</tok>
<tok base="respond" class="word" ctag="VBZ" id="t75346" msd="V,PRES,S,finite" sp="y">responds</tok>
<tok base="to" class="word" ctag="TO" id="t75347" msd="PP" sp="y">to</tok>
<tok base="choice" class="word" ctag="NNS" id="t75348" msd="N,PL" sp="y">choices</tok>
<tok base="make" class="word" ctag="VBN" id="t75349" msd="V,PAST,ED,finite" sp="y">made</tok>
<tok base="by" class="word" ctag="IN" id="t75350" msd="PP" sp="y">by</tok>
<tok base="program" class="word" ctag="NN" id="t75351" msd="N,SG,proper,vrbl" sp="y">program</tok>
<tok base="user" class="word" ctag="NNS" id="t75352" msd="N,PL" sp="y">users</tok>
<tok base="." class="punc" ctag="." id="t75353" msd="" sp="y">.</tok>
</s>
```

Figure 4.2: The sentence *"Simulation program: A computer program that simulates an authentic system (city, pond, company, organism) and responds to choices made by program users."* in XML format

described as a formal definition of objects).

Besides these three definition types, a fourth category was also introduced for those definitions that are not captured under any of the other three types or where some fundamental component is missing (`other_def`). For example, the sentence *Browsers, tools for navigating the Web, can also reproduce sound* belongs to this fourth category because the *definiens*, *tools for navigating the Web*, is given by means of an apposition statement.

Some definitions span more than one sentence. In some cases, the *definiendum* is present in the first sentence and is missing in the subsequent ones. In other cases, the *definiendum* is introduced in a previous sentence and is referred to again in the subsequent ones, where the *definiendum* is replaced by a pronoun.

This fourth type of definitional sentences are not accounted in the present research, but was covered by the manual annotation of the corpus in order to determine to which

extent they are present, and to allow future work to take in consideration this kind of sentences.

Accordingly, the definition typology is made up of four different classes whose members were tagged with `punct_def`, for definitions whose connector is a punctuation mark, with `is_def`, for copula definitions, `verb_def`, for verbal definitions, and finally `other_def`, for all the remaining definitions.

All the components of each definitional sentence were explicitly distinguished. This involves the explicit markup of the sentence containing the definition (`definingText` tag), the *definiendum* (`markedTerm` tag), the connector (`connector` tag) and also the type of definition (`defType1` attribute). Figure 4.3 shows an example of the mark-up of a definitory context.

This annotation was performed by three different human annotators.

```
<par id="p197" rend="P class=MsoNormal style=text-align:justify">
 <s id="s329">
  <definingText def="m252" def_type1="is_def" id="d25">
   <markedTerm dt="y" id="m252">
    <tok base="ftp" class="word" ctag="PNM" id="t10354" sp="y">FTP</tok>
   </markedTerm>
   <connector>
    <tok base="ser" class="word" ctag="V" id="t10355" msd="pi-3s" sp="y">é</tok>
   </connector>
    <tok base="um" class="word" ctag="UM" id="t10356" msd="ms" sp="y">um</tok>
    <tok base="protocolo" class="word" ctag="CN" id="t10357" msd="ms" sp="y">protocolo</tok>
    <tok base="que" class="word" ctag="CJ" id="t10358" sp="y">que</tok>
    <tok base="possibilitar" class="word" ctag="V" id="t10359" msd="pi-3s" sp="y">possibilita</tok>
    <tok base="a" class="word" ctag="DA" id="t10360" msd="fs" sp="y">a</tok>
    <tok base="transferência" class="word" ctag="CN" id="t10361" msd="fs" sp="y">transferência</tok>
    <tok base="de" class="word" ctag="PREP" id="t10362" sp="y">de</tok>
    <tok base="arquivo" class="word" ctag="CN" id="t10363" msd="mp" sp="y">arquivos</tok>
    <tok base="de" class="word" ctag="PREP" id="t10364" sp="y">de</tok>
    <tok base="um" class="word" ctag="UM" id="t10365" msd="ms" sp="y">um</tok>
    <tok base="local" class="word" ctag="CN" id="t10366" msd="ms" sp="y">local</tok>
    <tok base="para" class="word" ctag="PREP" id="t10367" sp="y">para</tok>
    <tok base="outro" class="word" ctag="ADJ" id="t10368" msd="ms" sp="y">outro</tok>
    <tok base="por_" class="word" ctag="PREP" id="t10369">por_</tok>
    <tok base="a" class="word" ctag="DA" id="t10370" msd="fs" sp="y">a</tok>
    <tok base="internet" class="word" ctag="CN" id="t10371" msd="fs">Internet</tok>
    <tok class="punctuation" ctag="PNT" id="t10372" sp="y">.</tok>
 </definingText>
 </s>
 </par>
```

Figure 4.3: The sentence *FTP é um protocolo que possibilita a transferência de arquivos de um local para outro pela Internet. (FTP is a protocol that allows the transfer of files from one location to another on the Internet)* in final XML format

Type	Infor. Society	Infor. Technology	e-Learning	Total
Punctuation	4	68	8	80
Copula	60	47	14	121
Verbal	39	33	26	98
Others	30	54	23	107
total	199	295	157	651

Table 4.2: The distribution of the different types of definitions in the Portuguese corpus

Table 4.2 displays the distribution of the different types of definitions in the Portuguese corpus. The domains of Information Society and Information Technology present a higher number of definitions, in particular copula definitions. This is due to the fact that most documents belonging to these domains were conceived to serve as tutorials for non-experts, and have thus a more didactic style. The part of the corpus concerning the e-learning domain is mostly composed by research papers and disertations, where the goal is less didactic. This distribution confirms the observations of Pearson (1998) reported in Chapter 2.2.2, that is, documents conceived for non-expert users present a higher number of definitions.

Table 4.3 reports the composition of the English corpus. In this case, it is not possible to make a clear distinction in sub-corpora, as all the documents cover the generic domain of information technology.

Punctuation	Copula	Verbal	Others	Total
130	112	124	56	422

Table 4.3: The distribution of the different types of definitions in the English corpus

4.3 The Rule-Based System

In order to take advantage of the XML format of the corpus, a regular expression based tool for pattern matching that was XML aware was convenient. The tool opted was LXtransduce[1]. It is a transducer which adds or rewrites XML markup on the basis of the rules provided. Lxtransduce is an updated version of fsgmatch, the core program of LTG's Text Tokenisation Toolkit (LT TTT) (Grover *et al.*, 2000). The LT TTT

[1] http://www.ltg.ed.ac.uk/~richard/ltxml2/lxtransduce-manual.html

is a software system which provides tools to perform text tokenisation and mark-up, developed within an XML processing paradigm whereby tools are combined together in a pipeline allowing each to add, modify or remove some piece of mark-up.

LxTransduce allows the development of grammars containing a set of rules, each of which may match part of the input. In case there is a successful match of the rule with some part of the input, it is possible to replace the matched text or wrap it with an xml tag. Rules may contain simple regular-expressions, or they may contain references to other rules in sequences or in disjunctions, hence making it possible to write complex procedures on the basis of simple rules. The outcome of a rule may be used to instantiate variables that can be used later on by other rules. A grammar is thus composed of several rules, where a main rule calls the remaining ones.

In general, the grammar is composed of simple rules for matching basic expressions, such as conjunctions, articles, or nouns. These rules are combined in order to feed more complex ones, aiming at matching complex structures as, for instance, noun phrases. As expected, special focus is given to the definition connector and syntactic patterns surrounding it.

In order to develop a heuristic for the extraction of definitions, a baseline grammar was built. This grammar marked as a definition every sentence containing the specific connector, which in the case of punctuation definitions, addressed in the present Chapter, is made up of the colon ":" and the dash "-".

A list of candidate definitions is extracted, containing all the definitions marked by the baseline grammar. By analyzing this list, it is possible to observe the lexical and syntactical patterns that characterize definitional and non-definitional sentences and consequently arrive at a set of rules to discard bad candidate definitions. Our aim is to elaborate a heuristic for the construction of such rules.

A development corpus, consisting of 75% of the whole Portuguese corpus, was inspected in order to draw generalizations helping to concisely delimit lexical and syntactic patterns entering definitory contexts. This sub-corpus was used also for testing the successive development versions of the grammar. The held out 25% of the corpus was not used in the development phase. It was reserved for testing the system and to obtain the evaluation results.

In order to evaluate the rule-based module, the usual measures of Recall and Precision were used, and also the F-measure, that combines the first two. These scores were

calculated at the sentence level: a sentence is considered a true positive of a definition if it contains a definition. Recall is the proportion of the sentences correctly classified by the system with respect to the sentences (manually annotated) containing a definition. Precision is the proportion of the sentences correctly classified by the system with respect to the sentences automatically selected as definitions.

4.4 Punctuation Definitions

All the sentences where one of the punctuation marks colon ":" and dash "-" were present were collected, making a total of 1519 such sentences, of which only 108 are definitions. This means that only 1 in each 14 sentences is a correct definition. This represents our system baseline, characterized thus by a precision value of 0.07, recall 1, and F-measure 0.13.

We inspected the structure of true positives (the definitions manually annotated) and the false positives (the sentences erroneously annotated). The focus was on the sentence left side, that is on the initial part of the sentence before the connector. The information taken into consideration was the part of speech (POS) and the lemma. For each one of these two pieces of information, two lists of partial sentences were collected, one with positive examples, the other with negative ones.

In this way, the relevant information to distinguish between definitions and non-definitions was made easy to identify. In particular, definitions, in comparison with non-definitions, typically present before the connector:

- a limited number of words;
- the absence of verbs in its indicative tense;
- the absence of a pronoun or a preposition starting the sentence;
- the presence of at least a noun;
- the absence of multiple occurrences of the same connector, that is the punctuation mark, in the sentence;

After the first phase of implementation of such rules, it was noted that a significant number of false positives occurred in indexes and in the bibliography sections present in some documents, such as articles and dissertations. In order to avoid to consider information included in these sections as candidate definitions, a set of heuristics was

developed in order to skip these sections, and to avoid those titles containing the target punctuation that could be considered definitions.

Eventually, it was noted that there are some nouns followed by target punctuation marks that must not be considered definitions, such as: *fax: 219948458* or *e-mail: rosa@di.fc.ul.pt*.

In order to prevent this kind of information from being considered a definition, a list of nouns was created containing words such as fax, e-mail, address, name, surname, mobile, etc. A rule was created stating that when the left size of the definition is composed by only a noun and this noun is in this list, the sentence is not considered a definition.

In Table 4.4, results for this pattern-based module are presented for the developing corpus, and the testing corpus.

	Portuguese		
	Precision	Recall	F-measure
Train	0.95	1	0.98
Test	0.87	1	0.93

Table 4.4: Results obtained by the rule-based module for punctuation definitions in Portuguese

4.4.1 Testing with the English Corpus

In order to further test the set of rules here developed, the rule-based module was applied to the English corpus. The rules were adapted to the English POS tag-set, and the list of words translated to English. Table 4.5 shows these results.

The results obtained are quite similar to the ones obtained with the test sub-corpus of Portuguese. This shows that the rule-based module is sufficiently robust to be applied to different languages without a worsening of its performance. In regards to the

	English		
	Precision	Recall	F-measure
Test	0.82	1	0.90

Table 4.5: Results obtained by the rule-based module for punctuation definitions in English

punctuation definitions, not only a method for obtaining the rules that is easily applicable to other contexts, but the rules themselves, have been created so that they can be effectively translated and applied to other languages.

4.4.2 Discussion and Error Analysis

Comparing these results against those by other works is, as usual, not straightforward, as only a few works focus on this type of definitions. The three cases addressed in Chapter 3 report results far below the ones obtained here. In particular, Westerhout & Monachesi (2007) and Iftene *et al.* (2008) achieved an F-measure of 0.05 and 0.26 respectively.

It is interesting to compare the results obtained here for English with the ones reported by Borg *et al.* (2009) as in our work a sample of the corpus was used, so the performances, in this case, are almost directly comparable. These authors report a precision of 0.33, a recall of 0.12 and an F-measure of 0.17. As far as we know, this author did not develop rules for dealing with bibliography sections present in some documents, and they do not mention any kind of lists containing specific words to help to filter non definitions. When we were testing our extractor before introducing these rules, our results were quite similar to the ones reported by Borg *et al.* (2009).

Analyzing the false positives for both languages, that is sentences that were erroneously recognized as definitions, it is possible to group them in a few categories.

Regarding the Portuguese corpus, the final version of the pattern-based module returned only 8 false definitions, 4 in the training corpus and 4 in the testing corpus, while for the English corpus, 32 false definitions were returned. Of these, 3 sentences from the Portuguese corpus and 11 from the English one are made up of section titles that the module for detecting indexes in the document was not able to detect and avoid. This is because this module detects the index section of a document and excludes all the sentences that are identical to a title in the index. This cannot be applied to documents without an index section but that still are organized in sections. Examples are:

- *E-learning: reflexões em torno do conceito*
- *Understanding Distance Education - a framework for the future*

These false positives could be avoided if the original documents retained layout information. In this case, it would be very easy to implement rules escaping sentences if these were formatted as titles.

A second group of false positives is related to a list of possible qualities or values that a certain object could have, such as for example:

- *Optional fields: volume or number, series, type, address, edition, month.*
- *Soluções humanas: caneta de ponta de feltro, de realce e bloco de notas, em pontos de informação.*

A third group was found only in the English corpus. These sentences provide some observations regarding a term in a specific context. See, for example:

- *the cost – interactive displays tend to be more expensive than static ones*;
- *software and hardware – software and hardware is needed for both editing and presentation*;
- *equipment – extra equipment may be needed to produce and use the programs, including speakers, video recorders and players, CD-ROM recorders and players, touch screens, mouse alternatives, etc.*;
- *the audience – it is difficult to produce interactive programs which will appeal to all ages and types of user*;

Arguably, these examples in the last two groups represent a limit which is difficult to surpass without performing a deeper analysis of the text, taking into consideration the semantic and pragmatic dimension of a document.

4.5 Baselines for the Other Definition Types

Similar heuristics to develop rules in a semi-automatic way were also applied to the other two types of definitions, with a double purpose. First, the aim was to get a baseline against which we will be able to compare the results obtained with more advanced methods. Second, we want to compare these results with the state of the art.

4.5.1 Copula Definitions

In order to develop a baseline module for the extraction of copula definitions, a baseline grammar was built. This grammar marked as a definition every sentence containing the verb *to be* as its main verb. This way, a list composed of 1,360 sentences was extracted, of which 113 are actual definitions. This means that in terms of evaluation measures, we obtained a recall of 1, a precision of 0.08, and an F-measure of 0.15.

Analyzing the list of resulting candidate definitions, it is possible to observe the lexical and syntactical patterns that characterize definitions and non-definition sentences and consequently derive a set of rules to extract definitions. The analysis of the sentences was carried out in a semi-automatic fashion. In order to support this grammar with syntactic information, the copula definitions manually marked in the developing corpus were gathered. The focus was on characterizing the part-of-speech (POS) patterns after the connector (and not before as in punctuation definitions), as after observation, it resulted to be the more informative. For this reason, all linguistic information was removed except the information regarding POS in order to highlight relevant patterns. This means that for each `tok` in the XML format, only the value of `ctag` attribute is taken in consideration. In Figure 4.4, the relevant information is made evident.

In order to get relevant patterns, only context windows composed of 3, 4 and 5 tokens after the connector were considered. This way, we ended up with a list of POS patterns after the connector. Patterns occurring more than three times in the definitions annotated in the development corpus were implemented in this grammar.

Eventually, the syntactic patterns of all sentences erroneously marked as definition by the baseline grammar were extracted and analyzed using the same procedure. This way we got patterns that were common to definitory and non-definitory constructions. Patterns whose occurrence was higher in the erroneously marked definitions than in the manually marked ones were not implemented.

The final grammar for copula definitions was composed of 53 rules, of which 37 are simple rules (capturing nouns, adjectives, prepositions, etc.), 5 are rules for capturing the verbal part, 9 are for nominal and prepositional phrases, and 2 are rules for capturing the definitory context.

The following rule is an example of the rules in the copula grammar:

```
<par id="p197" rend="P class=MsoNormal style=text-align:justify">
 <s id="s329">
  <definingText def="m252" def_type1="is_def" id="d25">
   <markedTerm dt="y" id="m252">
    <tok base="ftp" class="word" ctag="PNM" id="t10354" sp="y">FTP</tok>
   </markedTerm>
   <connector>
    <tok base="ser" class="word" ctag="V" id="t10355" msd="pi-3s" sp="y">é</tok>
   </connector>
    <tok base="um" class="word" ctag="UM" id="t10356" msd="ms" sp="y">um</tok>
    <tok base="protocolo" class="word" ctag="CN" id="t10357" msd="ms" sp="y">protocolo</tok>
    <tok base="que" class="word" ctag="CJ" id="t10358" sp="y">que</tok>
    <tok base="possibilitar" class="word" ctag="V" id="t10359" msd="pi-3s" sp="y">possibilita</tok>
    <tok base="a" class="word" ctag="DA" id="t10360" msd="fs" sp="y">a</tok>
    <tok base="transferência" class="word" ctag="CN" id="t10361" msd="fs" sp="y">transferência</tok>
    <tok base="de" class="word" ctag="PREP" id="t10362" sp="y">de</tok>
    <tok base="arquivo" class="word" ctag="CN" id="t10363" msd="mp" sp="y">arquivos</tok>
    <tok base="de" class="word" ctag="PREP" id="t10364" sp="y">de</tok>
    <tok base="um" class="word" ctag="UM" id="t10365" msd="ms" sp="y">um</tok>
    <tok base="local" class="word" ctag="CN" id="t10366" msd="ms" sp="y">local</tok>
    <tok base="para" class="word" ctag="PREP" id="t10367" sp="y">para</tok>
    <tok base="outro" class="word" ctag="ADJ" id="t10368" msd="ms" sp="y">outro</tok>
    <tok base="por_" class="word" ctag="PREP" id="t10369">por_</tok>
    <tok base="a" class="word" ctag="DA" id="t10370" msd="fs" sp="y">a</tok>
    <tok base="internet" class="word" ctag="CN" id="t10371" msd="fs">Internet</tok>
    <tok class="punctuation" ctag="PNT" id="t10372" sp="y">.</tok>
 </definingText>
 </s>
 </par>
```

Figure 4.4: The sentence *FTP é um protocolo que possibilita a transferência de arquivos de um local para outro por a Internet. (FTP is a protocol that allows the transfer of files from one location to another on the Internet)* with `ctag` information highlighted

```
<rule name="copula1">
<seq>
<ref name="SERdef"/>
<best> <seq>
<ref name="Art"/>
<ref name="adj|adv|prep|" mult="*"/>
<ref name="Noun" mult="+"/> </seq>
<ref name="tok" mult="*"/>
<end/> </seq> </rule>
```

This is a complex rule that makes use of other rules previously defined in the grammar covering copula definitions. This rule matches a sequence composed by the verb *to be* followed by an article and one or more nouns. Between the article and the noun, an

adjective, an adverb or a preposition can occur.

4.5.2 Verbal Definitions

In order to develop a grammar for definitions based on verbal definitions, a slightly different approach was used. As a first step, all the verbs marked as connectors were extracted, obtaining a list of 37 verbs. This list was then improved by adding synonyms, ending up with a final list of 46 different verbs. All sentences containing these verbs as the main verb were extracted, resulting in 744 sentences, of which 98 were effective definitions, leading to a precision of 0.13 and an F-measure of 0.23.

For each sentence, the POS information after the connector was extracted, thus yielding a list of definitory patterns. Differently from copula definitions, this time the verb form information was retained in order to highlight the syntactic behavior for each verb. This way it was possible to divide verbs into 4 different categories: verbs that appeared in the active form, verbs that appeared in the passive form, verbs that appeared in the reflexive form, and verbs that were followed by a preposition. This information was listed in a separate file called *lexicon*. For each class a rule was written. Table 4.6 shows the list of these verbs with the corresponding classification.

The following rule is a sample of how verbs are listed in the lexicon.

```
<lex word="significar"> <cat>act</cat> </lex>
```

In this example the verb *significar* ("to mean") is listed in its infinitive form, that corresponds to the attribute `base` in the corpus. The tag `cat` indicates a category of the lexical item. In our grammar, `act` indicates that the verb occurs in definitions in the active form. A rule was written to match this kind of verb:

```
<rule name="ActExpr">
<query match="tok[mylex(@base) and (@msd[starts-with(.,'fi-3')]
or @msd[starts-with(.,'pi-3')])]"constraint="mylex(@base)/cat='act'"/>
<ref name="Adv" mult="?"/>
</rule>
```

This rule matches a verb in the present or future tense (third person singular and plural) but only if the base form is listed in the lexicon and the category is equal to `act`. Similar rules were developed for verbs belonging to the other categories. The final grammar is composed of 65 rules: 42 simple rules, 10 rules capturing verbs, 9 for matching nominal and prepositional phrases, and 4 rules for capturing the definitory context.

VERB		REFLEXIVE	ACTIVE	PASSIVE	PREPOSITION
abarcar	to cover		x	x	
abordar	to approach		x	x	
abranger	to include		x	x	
assentar	to based				x
associar	to associate			x	
assumir	to assume	x		x	
basear	to base	x		x	
caracterizar	to characterize	x	x	x	
chamar	to call	x		x	
classificar	to classify	x	x	x	
compor	to compose	x	x	x	
conhecer	to know	x		x	
considerar	to consider	x		x	
consistir	to consist		x	x	
constar	to consist		x		
constituir	to constitute	x	x	x	
corresponder	to correspond		x		
definir	to define	x	x	x	
denominar	to call	x	x	x	
descrever	to describe	x	x	x	
designar	to designate	x	x	x	
destinar	to intend	x	x	x	
englobar	to encompass		x	x	
envolver	to involve				x
especificar	to specify	x	x	x	
estabelecer	to establish		x		
explicar	to explain	x	x	x	
expressar	to express	x	x	x	
formar	to form			x	
fundar	to be based	x		x	
identificar	to identify	x	x	x	
implicar	to imply				x
incluir	to include				x
indicar	to indicate				x
permitir	to allow				x
perspectivar	to perspective	x	x	x	
possibilitar	to enable				x
preconizar	to profess	x	x	x	
providenciar	to arrange				x
referir	to refer	x		x	
reportar	to concern	x		x	
representar	to represent	x	x	x	
significar	to mean		x		
tratar	to treat	x			x
visar	to aim				

Table 4.6: List of the definitional verbs used in the rule-based module for verbal definitions

Portuguese copula definitions results			
	Precision	Recall	F-measure
Test	0.32	0.66	0.43
Train	0.30	0.69	0.42
Total	0.31	0.68	0.42

Table 4.7: Results obtained by the rule-based module for copula definitions in Portuguese

Portuguese verbal definitions Results			
	Precision	Recall	F-measure
Test	0.14	0.65	0.23
Train	0.12	0.73	0.21
Total	0.13	0.68	0.22

Table 4.8: Results obtained by the rule-based module for verbal definitions in Portuguese

4.5.3 Evaluation

Table 4.7 displays the results of the copula grammar, for each sub-corpora and for the full corpus.

Regarding the results obtained with the test sub-corpus, if compared with other systems adopting the same definition typology, though involving a more intense effort in generating the rules, as for example Westerhout & Monachesi (2007) and Borg *et al.* (2009), the performance of heuristic presented here lies above the state of the art. With an F-measure of 0.43, the grammar outperforms both cited works, which report an F-measure score of 0.34 and 0.26 respectively.

Table 4.8 in turn shows the results we obtained with the grammar for verbal definitions. Comparing these results to those obtained with the copula grammar, we notice that they are less satisfactory. This is probably due to the larger diversity of patterns for each verb. If compared with the two works cited for copula definitions, the F-measure value is lower. But if we consider the recall value of 0.65, our grammar outperforms those of Westerhout & Monachesi (2007) and Borg *et al.* (2009), with a recall of 0.41 and 0.32, respectively.

Table 4.9 presents the scores obtained by a grammar that combines all the three sub-grammars described in this work. This table displays the overall performance of

Portuguese Global Results			
	Precision	Recall	F-measure
Total	0.41	0.75	0.53

Table 4.9: Results obtained by the rule-based module for definitions in Portuguese

the system based on the grammars developed so far. Any sentence that is tagged as a definitory context (no matter which definition type it receives) will be brought on board. The recall value remains quite high, 0.75, while it is clear that for the precision value, 0.53, there is still much room for improvement.

When compared to other works that adopt rule-based grammars, we can state that the heuristics developed here are in line with the state-of-the-art results or, in some cases, they outperform it.

4.5.3.1 Extrinsic Evaluation

On a par with the quantitative evaluation, a qualitative evaluation was carried out by a group of users (Avelãs *et al.*, 2008). Six graduate students were presented with a list of definitions automatically generated by the definition extraction system just presented. The corresponding document was a 12-page introduction to the use of Internet, and our tool extracted 34 different definitions. The testers were instructed to read the document carefully and then to score each definition using a rating scale from 1 to 4 (very good definition, good definition but not complete, acceptable definition, not a definition at all). The average score was 2.21, thus indicating that the automatically extracted passages are on average considered accettable definitions according to human appreciation.

Moreover, six university tutors were presented with a document, and were requested to generate a glossary using the tool, or without it. Then, they were asked to compare the different experience they had had in terms of time spent accomplishing the task, and satisfaction using the extractor. All testers agreed that the tool is useful, although some of them thought that the tool could be improved, and said they would use it if available (Del Gaudio & Branco, 2009a).

4.6 Conclusions

In this Chapter the corpora used for developing a series of heuristics for automatic definitions extraction were presented. The type of annotation and definitions were stated as well as a set of simple heuristics to develop a grammar for each definition typology.

This work has shown that is is possible to build rules for the extraction of definitions in a semi-automatic fashion, focusing mostly on syntactic information around the connector. The creation of lists consisting of this information can quickly highlight the relevant information that may characterize the definitory sentences. This way, it is possible to achieve good results, that are comparable to the state of the art for all three definition types.

Results of these first heuristics represent the baseline for the works present in the next Chapters.

In addition, this Chapter described the first step of what we call the "divide and conquer" approach. When punctuation definitions are dealt with this method, results outperforming the state of the art are achieved, not only in terms of performance, but also in terms of portability. In fact, the rules have proved to be effective enough to be adapted to a corpus in another language, with just a slightly worsening in performance. This portability is a great achievement in the automatic extraction of definitions.

Chapter 5

Copula Definitions: Machine Learning Approach

5.1 Introduction

In this Chapter[1] the second step of the "divide and conquer" approach is described, focusing on copula definitions. As the rule-based module is not effective when applied to this kind of definitions, a different methodology was carried out based on machine learning techniques.

The definition extraction problem can be envisaged as a binary classification task, where each sentence should be assigned the correct class, i.e. whether it is a definition. In a corpus of naturally occurring texts, it typically happens that the number of sentences expressing a definition is much smaller than the number of sentences that are not definitions. This gives rise to imbalanced datasets that, depending on the corpus, may present different degrees of imbalance, which nevertheless tends to be always quite high. For example, using a corpus consisting of encyclopedic texts (that are usually rich in definitions) and web documents (less rich in definitions), Tjong *et al.* (2005) report that only 18% of its sentences contained definitions. For this reason, this methodology requires the handling of datasets with sparse evidence for the class of interest, representing a case study of coping with highly imbalanced datasets in Natural Language Processing.

Research on automatic definition extraction has made use of sampling techniques only very marginally. To a large extent, this is due to the fact that the extraction

[1]Part of this Chapter was published as Del Gaudio *et al.* (2013)

of definitions is performed by applying pattern-matching rules first. Machine learning techniques are subsequently applied to improve the outcome of the pattern-matching module, whose previous application had already reduced the imbalance of the dataset.

The drawback in this methodology to address the definition extraction task is that the pattern-matching modules are typically specific for a particular domain and, in any case, always specific for a particular language. By eliminating the pattern-based step and directly applying machine learning algorithms, it is possible to overcome the limitations imposed by that methodology. And it is, thus, in this scenario that the imbalanced dataset issue needs to, and can be, tackled.

In order to support the idea that this methodology is applicable to different contexts, the experiments on copula definitions were carried out using not two, but three corpora in different languages, that is, beside Portuguese and English corpora, a third corpus in Dutch was also used.

With the research presented in this Chapter, we seek to contribute to the advancement of the task of definition extraction by exploring the methodology that addresses it only by means of machine learning techniques. More generally, as this methodology requires the handling of datasets with sparse evidence for the class of interest, this work offers a case study of coping with highly imbalanced datasets in Natural Language Processing (NLP).

Next Section presents an overview of the general problem of classification when datasets are imbalanced and a more specific overview on imbalanced dataset in NLP and in particular in the field of definition extraction. The aim here is to describe how this issue is addressed in different areas, and then bring the focus to NLP. It also describe the best metrics to perform evaluation. Section 5.3 presents the experimental settings of this work, in particular the datasets used, the learning and sampling algorithms and their combination. The results achieved with the different combination of algorithms are reported in Section 5.4. These results are discussed in Section 5.5. Section 5.6, present and discuss the results obtained with the same methodology applied to verbal definitions. Finally, Section 5.7 presents the conclusions that can be drawn on the basis of the work carried out.

5.2 The Imbalanced Data Issue

The issue of training classifiers with imbalanced data emerges in different real-world application domains where, for different reasons, the minority class is the one of interest, such as financial fraud detection (Bay *et al.*, 2006), disease diagnoses (Taft *et al.*, 2009), or malicious network activity detection (Vatturi & Wong, 2009). The imbalance can be quite dramatic, from a ratio of 1 to 100 to even of 1 to more than 10,000 (Wu & Chang, 2003).

As pointed out by Chawla *et al.* (2004), when common classification algorithms are trained with and applied to such skewed data, they tend to be overwhelmed by the majority classes and ignore the minority ones. This occurs because in most classification learning algorithms, the objective is to minimize the overall classification error and this does not account for classification error on each individual class. It happens that, for example, by using a dataset with a ratio of 1 to 10, a classifier achieves approximately 90% accuracy just by always predicting the majority class.

A variety of solutions to the class-imbalance problem have been proposed that lend themselves to be grouped under the following major approaches: to rebalance the dataset; to apply a cost to classification errors; or to modify the learning algorithms to make them more suitable to address this issue.

In general, a common practice for dealing with imbalanced datasets is to rebalance them artificially, by either over-sampling the minority class or under-sampling the majority class. This includes random over-sampling, random under-sampling, directed over-sampling (in which minority class examples are replicated, but the choice of samples to replicate is informed rather than random), directed under-sampling (where, again, the choice of examples to eliminate is informed), over-sampling with informed generation of new synthetic samples (such as SMOTE), and combinations of the above techniques.

Determining which sampling method is the best greatly depends on the chosen classifier and the properties of the application, including how the samples are distributed in the multidimensional space or the extent to which the different classes are mixed. Therefore, a systematic investigation of different sampling approaches is important and required to optimize the performance of the system at stake.

A different solution is to adjust the costs of the various classes so as to counter the class imbalance. As the cost of misclassifying a minority-class example is greater than the cost of misclassifying a majority-class example, it is possible to take the misclassification costs into consideration in order to minimize the overall misclassification cost. For highly skewed class distributions, this allows the classifiers to not always predict the majority class and helps them to perform better on the minority class than if the misclassification costs were equal. A drawback of this approach is that it usually assumes that the costs of making an error can be known (Elkan, 2001; Ling & Sheng, 2008), which is not always the case. Additionally, in a comparative study assessing over-sampling, under-sampling and cost-sensitive approaches, no relevant difference was found (Weiss *et al.*, 2007). Additionally, there is no guarantee that the distribution of examples in the dataset used to create a classifier is the same as the one of the testing data, which may even be worsened when that classifier is applied to other examples.

5.2.1 Evaluation Issues

When dealing with datasets with a high degree of imbalance, two commonly used metrics to assess the performance of classifiers, accuracy and error rate, consider different classification errors as equally important, an assumption that is hardly true in imbalanced data domains. Misclassifying minority class examples is frequently much more critical and costly than the opposite, as discussed in the previous section. For instance, in medical diagnosis, the error of diagnosing a sick patience as healthy (misclassifying an item from the minority class) is considered a serious error while the opposite is considered much less critical. As a consequence, these metrics are biased to "favor" the majority class. In a dataset dominated by a majority class, a simple way of maximizing accuracy (or minimizing error rate) is to correctly classify the majority class examples. This issue can be clearly seen by a trivial classifier that classifies every example as belonging to the majority class, and therefore makes no incorrect classifications on this class. In a 90% majority class dataset, such a (useless) classifier is able to achieve 0.90 accuracy (or 0.1 error rate), even though it misclassifies every minority class example.

Given these difficulties, it is recommendable to use metrics different from accuracy or error rate, or at least not to rely solely on these metrics.

The F-measure is a popular performance measure in text classification and information retrieval applications. Such applications are often characterized by large class

imbalances and having a minority class of more interest than the majority one. F-measure gauges the performance of the class of interest (the positive class, usually the minority class) by measuring its *precision* and *recall*, and composing both by a harmonic mean. Although it is a popular measure, the precision component of F-measure is dependent on the class distribution[1] (Prati *et al.*, 2011), and therefore, it must be assumed that the class distribution is fixed. A practical problem arises when comparing classifiers for a similar problem, but generated over datasets with different class distributions. It is difficult, if not impossible, to fully characterize how much of the performance differences among the classifiers are due to differences of the techniques and how much was caused by an increase/decrease in precision due to differences in class distribution.

There is a need for a metric as much independent from data set structure as possible. Only in this way, it can be expected that results obtained with a new data set are comparable with the ones used to train the classifiers. Batista *et al.* (2004) dealt with this issues and reported four metrics that are not sensitive to the class distribution in a data set.

These metrics are:

- False Negative rate: the percentage of positive examples misclassified as belonging to the negative class
$$FN_r = \frac{FN}{(TP + FN)}$$
- False Positive rate: the percentage of negative examples misclassified as belonging to the positive class
$$FP_r = \frac{FP}{(FP + TN)}$$
- True Negative rate: the percentage of negative examples correctly classified as belonging to the negative class
$$TN_r = \frac{TN}{(FP + TN)}$$

[1] An intuitive argument is that it is easy to obtain high precision in domains in which the prevalence of positives is also high.

- True Positive rate: the percentage of positive examples correctly classified as belonging to the positive class

$$TP_r = \frac{TP}{(TP + FN)}$$

According to Batista a good classifier should try to minimize FN and FP rates, and maximize TN and TP rates. Unfortunately, there is a trade-off between these two metrics, and in order to analyze this relationship, ROC graphs are used (Fawcett, 2004).

Figure 5.1 shows an example of ROC graphs. ROC graphs are two-dimensional graphs where TP rate is plotted on the Y axis and FP rate is plotted on the X axis. ROC graphs are consistent for a given problem even if the distribution of positive and negative instances is highly skewed.

In these graphs, the lower left point (0, 0) represents the strategy of never issuing a positive classification: such a classifier produces no false positive errors but also gains no true positives. The opposite strategy, of unconditionally issuing positive classifications, is represented by the upper right point (1, 1).

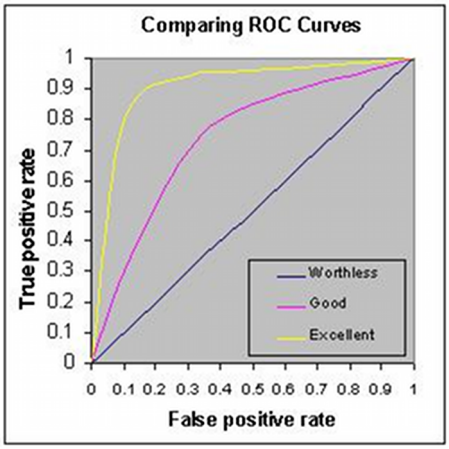

Figure 5.1: ROC graphs: an example

In order to assess the performance of a classifier, it is possible to reduce the respective ROC curve to a scalar value representing its performance. That is the Area Under the ROC curve (AUC), which is a portion of the area of the unit square. Its value will always be between 0 and 1. As random guessing produces the diagonal line between

(0,0) and (1,1), which has an AUC of 0.5, no realistic classifier should have a score lower than 0.5 under this metric.

5.2.2 The Imbalanced Dataset Issue in Natural Language Processing

The imbalanced data issue is a ubiquitous problem in Natural Language Processing given the Zipfian nature of many language phenomena and dimensions. In recent years, tasks such as sentence boundary detection (SBD), word sense disambiguation (WSD) or Named Entity Recognition (NER) have been addressed with machine learning techniques that explicitly seek to handle the imbalanced dataset issue.

As for the WSD task, the class imbalance issue arises due to the fact that word senses present a highly skewed distribution. To address this problem, Zhu (2007) adopted active learning with resampling methods. Active learning, also called optimal experimental design, is a semi-supervised machine learning, where a learning algorithm interactively queries an external information source to get the desired outputs at new data points. He tested random under- and over-sampling and an improved version of random over-sampling, called BootOS. In this case, each majority example has the same probability to be selected for the sampling, thus making the sampling not completely random. He found out that when the number of learned samples for each word was small, the BootOS has the best performance, followed by random over-sampling technique. As the number of learned samples increases, over-sampling and BootOS tend to support similar performances of classifiers in terms of accuracy and recall.

Regarding the NER task, Tomanek & Hahn (2009), as well as Zhu (2007), dealt with the imbalanced data problem in the context of active learning, having tested different approaches to reduce the imbalance. The objective of their work was to obtain more balanced datasets during annotation time by using active learning as a strategy to acquire training material. They applied over- and under-sampling techniques during active learning selection and after active learning iteration. In this last scenario, either examples for the minority class were over-sampled (e.g. by simple replication), or examples of the majority class were discarded to achieve a more balanced dataset. They concluded that under-sampling is disadvantageous when active learning is used due to the the fact that, after having spent human effort on labeling the selected sentences in an active learning iteration, some of these are immediately discarded in the next step. Over-sampling, in turn, entails computational overload.

In the SBD classification task, for each inter-word boundary, the goal is to identify it as either a sentence boundary or just a word boundary in the same sentence. As sentence boundaries are less frequent than non-sentence boundaries, it is necessary to deal here with an imbalanced dataset distribution. Liu *et al.* (2006) carried out a preliminary study using two corpora, made of conversational speech over the phone and broadcast news speech, where only about 13% of the inter-word boundaries corresponded to sentence boundaries in phone speech, and 8% in broadcasted speech. In this study, with performance measured using AUC, classifiers trained under the sampling approaches outperform those trained over the original training set. They also experimented with bagging, a meta-algorithm for combining different learning algorithms, which is a special case of model averaging, that can be used with any type of model for classification or regression. Bagging was found to significantly improve system performance for each of the sampling methods. They also reported the results of an empirical evaluation in a pilot study, showing that under-sampling the dataset works reasonably well and requires less training time. Over-sampling with replication increases training time without any gain in classification performance. SMOTE, a smart over-sampling method, outperforms the under-sampling approach when few features are used, but not when different combination of features are used. Bagging was also investigated on a randomly under-sampled training set, an ensemble of multiple under-sampled training sets, and the original training set. Bagging on an under-sampled training set versus the original training set without bagging results in an even better performance than the use of more samples.

Besides the detection of sentence boundaries, Liu *et al.* (2006) also investigated the detection of disfluency interruption points (the position at which point the speaker breaks off the original utterance and fluent speech becomes disfluent), taking in consideration the effect of different dataset size, sampling methods and learning methods. Regarding sampling methods they experimented with different options: no sampling, under-sampling, over-sampling, and an ensemble sampling, that split the majority class into N sets, each of which is combined with all of the minority class samples to make a balanced training set to train a classifier. Regarding learning methods, they tested, besides bagging, ensemble bagging and boosting. The former consists in the application of bagging on each balanced training set formed by the ensemble sampling approach. The latter combines multiple weak learning algorithms where each classifier is built based on

the output of the previous classifiers, mostly by focusing on the samples for which the previous classifiers made incorrect decisions. Results show that bagging benefits both tasks, but to different degrees. The benefit from ensemble bagging decreases as data size increases, and boosting can outperform bagging under certain conditions.

5.2.3 The Imbalanced Dataset Issue in Definition Extraction

In the literature, when machine learning techniques are applied to automatic definition extraction task, the output of a pattern-matching module are used as the training dataset. Since the pattern-based module is characterized by a good performance in terms of recall and poor performance in terms of precision, the machine learning module is used as a filter to discard false positive examples returned by the previous pattern-matching step (Fahmi & Bouma, 2006; Miliaraki & Androutsopoulos, 2004; Westerhout & Monachesi, 2008).

In very few cases, the machine learning algorithms were applied alone, skipping the pattern-based step and without facing the problem of data imbalance (Chang & Zheng, 2007).

The problem with the approach based on pattern-matching is that it relies strongly on the set of manually crafted rules developed to ensure the first step of the process. Excluding the case of a few very general heuristics, whenever one needs to build a system to extract definitions, it is necessary to start almost from scratch, by starting to analyze a possible set of definitions and then building a set of specific patterns. Furthermore, these rules are not only pertinent to a specific natural language, but also to a specific domain and application, making it difficult to extend their use beyond the constrained applicational context within which they were developed.

To the best of our knowledge, only three research papers have abandoned this widespread approach of starting with a first pass through a pattern-matching module, and sought to explicitly address the imbalanced dataset issue through some kind of sampling.

Degórski *et al.* (2008b) used a corpus in Polish composed of 10,830 sentences, 546 of which were definitions, with a ratio of 1:19.

By means of random under-sampling a balanced dataset was obtained and different machine learning classifiers were tested such as Naïve Bayes, C4.5, ID3, IB1, nu-SVC, AdaBoost with Decision Stump (AB+DS). The best result was obtained with the

AB+DS classifier with 0.18 of precision, 0.60 of recall and an F-measure score of 0.28. Instead of balancing the dataset before applying the classification algorithm, Kobyliński & Przepiórkowski (2008) applied directly the Balanced Random Forest algorithm, using the same number of items from minority and majority class, overtaking the issue of imbalanced datasets. In this way, a precision of 0.21 and recall of 0.69 were obtained.

Westerhout (2009) also applied Balanced Random Forest as a filtering module after a pattern-based module. These results were compared with the ones obtained by Naïve Bayes, also used as filter. In this experiment, Naïve Bayes and Balanced Random Forest showed very similar performance, with respectively, precision of 0.82 and 0.77, recall of 0.76 and 0.79, and F-measure of 0.79 and 0.78.

5.3 Experimental Setting

The main objective of the present work is to come up with a classifier for copula definitions, without the intermediation of a pattern-matching module. The main challenge of this approach is to find the best combination of sampling and learning algorithms for the construction of a classifier capable of recognizing copula definitions. In particular, we are interested in assessing the added value of applying sampling methods to handle the imbalanced dataset issue.

For assessing the portability of this work, we use three datasets derived from three different corpora, each one from a different language, namely Dutch, English and Portuguese.

The Dutch corpus presents characteristics similar to those of the two other corpora described in the previous Chapter. That is, it was collected in the context of the Language Technology for e-Learning LT4eL project, it covers the domains of Computer Science and e-Learning and is encoded in an XML-based format which includes the linguistic annotation of part-of-speech (POS), lemma and morphological analysis information (automatically assigned).

This corpus is composed of 26 tutorials with a total size of 353,174 tokens and 23,996 sentences, of which 113 contain copula definitions. The corpus was annotated with morphosyntactic features with the Wotan tagger and with lemmas provided by the CGN lemmatizer (Westerhout & Monachesi, 2007).

	Original Corpus		Sub Corpus		
	Token	Sentences	Sentences	Definitions	Ratio
Dutch	353,174	23,996	4,829	113	1:42
English	287,910	20,172	2,574	40	1:64
Portuguese	223,049	10,941	1,360	121	1:11

Table 5.1: Corpora description

In order to prepare the dataset to be used in the experiments, all the sentences where the verb "to be" appears as the main verb were extracted. For Portuguese, we obtained a sub-corpus composed by 1,360 sentences, 121 of which are definitions, with a ratio of about 1 to 11. For Dutch, we obtained a sub-corpus composed by 4,829 sentences, 120 of which are definitions, with a ratio of 1 to 42. Finally, for English, the sub-corpus is composed by 2,574 sentences, 40 of which are copula definitions, with a ratio of 1 to 64. These sub-corpora are datasets that were used to train and test the classifiers in the experiments reported below. Table 5.1 displays a quantitative description of all the three corpora in terms of their original size (tokens and sentences), dataset size, number of positive examples (actual definitions marked by the human annotators) and the ratio between positive and negative examples.

5.3.1 Feature Selection

The selection of features was determined by the goal of enhancing the transportability of the solutions for definition extraction, that is, we wanted the type of features used to be, as much as possible, domain and language independent.

Looking at related work, commonly used features are bag-of-words, n-grams (Miliaraki & Androutsopoulos, 2004) (either of part-of-speech or of base forms), the position of the definition inside the document (Joho & Sanderson, 2000), or the presence of determiners in the *definiens* and in the *definiendum*. Other relevant, more complex properties can be the presence of named entities (Fahmi & Bouma, 2006), or data from an external source such as encyclopedic data, Wordnet, etc. (Saggion, 2004).

Some of these features may work well with a given corpus but not so well with another. The use of the position of a definition-bearing sentence in its document is an example of a feature that is corpus dependent. For instance, in encyclopedic documents,

definitions appear at the beginning of documents, but the same did not happen in tutorials in our corpora.

In order to avoid such limitations, we represented instances as n-grams of part of speech (POS). Currently, a POS tagger is a basic resource available for many natural languages, which ensures that our approach is applicable to many languages.

After some preliminary experiments, the best choice turned out to be the adoption of bi-grams. From each corpora, we extracted all the bi-grams and then reduced the respective huge list using the Information Gain Attribute Evaluation algorithm (Witten & Frank, 2005). This algorithm evaluates features individually by measuring their information gain with respect to the classes. We obtained a list of 50 bigrams for each dataset.

5.3.2 Sampling Algorithms

In our evaluation, we selected a set of state-of-the-art sampling algorithms that are frequently used and referred to in the literature as delivering a good performance. We choose random (under and over) sampling algorithms as our starting point. We also selected the algorithms Condensed Nearest Neighbor Rule, Tomek Links, Edited Nearest Neighbor, Neighborhood Cleaning Rule as direct under-sampling methods, which are described below. In general, direct under-sampling methods try to characterize each training example as borderline, noise or far from the decision border, and they discard a subset of the examples according to this classification. For instance, noise examples may be discarded as well as a subset of the examples far from the decision border, since those examples are usually less critical (or harmful) for learning.

Most of the direct under-sampling algorithms were originally proposed as data cleaning methods. Therefore, they eliminate examples of both (minority and majority) classes. Unfortunately, for most imbalanced class datasets the number of minority class examples is severely small, and discarding part of those examples would often make the learning impracticable. Therefore, these data cleaning algorithms were adapted as under-sampling methods by simply retaining all minority class examples and applying the selection and filtering out over only the majority class examples.

We also use SMOTE as a direct over-sampling algorithm. SMOTE creates synthetic minority class examples instead of replicating exact copies of these examples.

A short description of each algorithm is presented.

Random over-sampling consists of random replication of minority class examples, while in **random under-sampling**, majority class examples are randomly discarded until the desired proportion is reached. These two straightforward methods are often criticized. Several authors have pointed out that under-sampling can potentially discard useful data that could be important for the induction process. In contrast, random over-sampling can increase the likelihood of overfitting, since it makes exact copies of the minority class examples (Batista *et al.*, 2005).

Condensed Nearest Neighbor Rule (CNN) (Hart, 1968) is a data cleaning method that finds a consistent subset in order to eliminate examples that are distant from the decision border, since these examples might be considered less relevant for learning. A subset $E' \subset E$ is consistent with E if using 1-nearest neighbor, E' correctly classifies the examples in E. An algorithm to find a consistent subset is: firstly, it randomly draws one example of each class from E and puts these examples in E'. Next, it uses a 1-NN algorithm over the examples in E' to classify each example in E. Every misclassified example from E is moved to E'. We converted this method into an under-sampling algorithm that preserves all instances of the minority class by adding to the subset generated by CNN all the minority examples it deleted. CNN is typically sensitive to noise, since noisy data is likely to be misclassified by its neighbors.

Tomek Links (Tomek, 1976) algorithm is a data cleaning method that removes both noise and borderline examples. Tomek Links are pairs of instances of different classes that have each other as their nearest neighbors. Given two examples E_i and E_j belonging to different classes, and $d(E_i, E_j)$ the distance between E_i and E_j, an (E_i, E_j) pair is called a Tomek Link if there is not an example E_k such that $d(E_i, E_k) < d(E_i, E_j)$ or $d(E_j, E_k) < d(E_i, E_j)$. If two examples form a Tomek Link, then either one of these examples is noise or both examples are borderline. As an under-sampling method, only examples belonging to the majority class are eliminated. The major drawback of Tomek Links is that this method can discard potentially useful data, since borderline examples are often important to characterize the decision border. This method has an higher order computational complexity and will run slower than the other algorithms.

Edited Nearest Neighbor Rule (ENN) (Wilson, 1972) is a data cleaning method, and it removes any example whose class label differs from the class of at least two of its three nearest neighbors. This algorithm was designed to identify and eliminate examples that are likely to be noise data, while retaining most of the data. Therefore, this

method is not very effective to balance training data. As an under-sampling method, we removed only majority class examples that disagree with their three nearest neighbors.

Neighborhood Cleaning Rule (NCL) (Laurikkala, 2001) is an under-sampling method that uses a variant of Wilson's Edited Nearest Neighbor Rule. NCL modifies the ENN rule in order to increase data cleaning. For a two-class problem, the algorithm can be described in the following way: for each example E_i in the training set, its three nearest neighbors are found. If E_i belongs to the majority class and the classification given by its three nearest neighbors contradicts the original class of E_i, then E_i is removed. If E_i belongs to the minority class and its three nearest neighbors misclassify E_i, then the nearest neighbors that belong to the majority class are removed.

While these methods are direct under-sampling techniques, SMOTE is an over-sampling method that produces new synthetic minority class examples.

SMOTE (Chawla *et al.*, 2002) is an over-sampling method that forms new minority class examples by interpolating between several minority class examples that lie together in the "feature space". For each minority class example, this algorithm introduces synthetic examples along the line segments joining any/all of the k minority class nearest neighbors (in this work k is equal to 3). Synthetic samples are produced by taking the difference between the feature vector (sample) under consideration and its nearest neighbors. The difference is multiplied by a random number between 0 and 1 and added to the feature vector under consideration.

SMOTE, random over and under-sampling are methods designed to change class proportion, and can be implemented to provide any desired output class distribution, including balanced distributions. The remaining methods are adaptations of data cleaning approaches and consequently they typically do not guarantee any desired class distribution. We investigated if multiple passes of the methods Tomek Links, ENN and NCL, would reach a perfect balance between positive and negative examples. This result was obtained only with Tomek Links. Even if we tried to force the other two to do so, they have a natural stop point, and depending on the nature of the size and the imbalance degree of the dataset, this stop point can occur before the balance is achieved.

All the described algorithms were first applied one by one and then coupled. In particular, we tried to pair under-sampling algorithms with over-sampling algorithms.

Three different settings were tested: under-sampling to 25%, 50% and 75% and subsequent over-sampling. This way, it is possible to assess to which extent a given algorithm is more effective.

Regarding ENN and NCL, when it was not possible to reach the desired class distribution (25%, 50% or 75%), we used the proportion returned by the under-sampling algorithm and then applied the over-sampling one to achieve the balance point.

5.3.3 Classification Algorithms

The selection of learning algorithms took into account two different considerations. Firstly, the selection of algorithms that represent the state of the art for definition extraction and also for imbalanced data. Secondly, the possibility to cover different paradigms of algorithms for classification, having at least an algorithm representative of each learning paradigm. This way, different sampling techniques may be studied with respect to a larger range of classification algorithms. Six such algorithms were selected: Naïve Bayes, C4.5, Random Forest, k-NN, Support Vector Machine (SVM) and Voting Feature Intervals.

A brief description of these algorithms can be found below:

Naïve Bayes (John & Langley, 1995) is a simple probabilistic classifier that is very popular in natural language applications. It is based on Bayes' theorem, and the algorithm is known for assuming independence of features. In short, the independence assumption means that the occurrence of a specific feature value is independent from the occurrence of any other feature value. In spite of its simplicity, it usually permits to obtain results similar to the results obtained with more sophisticated algorithms. Two different implementations were evaluated: one in which the numeric estimator precision values are chosen using a kernel estimator for numeric attributes and another one using a normal distribution. The latter obtained better overall performance, and for this reason only the results obtained with this configuration are presented.

C4.5 (Quinlan, 1996) and **Random Forest** (Breiman, 2001) are two decision tree algorithms. The first is a relatively simple algorithm that splits the data into smaller subsets using the information gain in order to choose the attribute for splitting the data. The second is an ensemble consisting of a collection of decision trees. For each tree, a random sample of the dataset is selected (the remaining is used for error estimation)

and for each node of the tree, the decision at that node is based on a restricted number of variables.

The k-**NN** algorithm (Aha *et al.*, 1991) is a type of instance-based learning, also called lazy learning because, unlike the algorithms above, the training phase of the algorithm consists only in storing the feature vectors and class labels of the training samples and all computation is deferred until the classification phase. In this phase, it computes the distance between the target sample and n samples in the dataset, assigning the most frequent class among the k nearest samples. As it is used by default in the literature, here we present results for the learner generated when k was set to 3.

The **SVM** algorithm tries to construct an N-dimensional hyperplane that optimally classifies data points as much as possible and separates the points of two classes as far as possible (Chang & Lin, 2001). The goal of SVM modeling is to find the optimal hyperplane that separates clusters of vectors in such a way that cases with one category of the target variable are on one side of the plane and cases with the other category are on the other side of the plane. The vectors near the hyperplane are the support vectors.

The **VFI** algorithm (Demiröz & Güvenir, 1997) uses a scheme that calculates the occurrences of feature intervals per class, and classifies by voting on new examples using these intervals, which are constructed around each class for each attribute. This way, an example is represented by a set of feature intervals on each feature dimension separately. Each feature participates in the classification by distributing real-valued votes among classes. Higher weight is assigned to more confident intervals, where confidence is a function of entropy. The class receiving the highest vote is declared to be the predicted class.

All classifiers were built using the Weka workbench (Witten & Frank, 2005). Regarding the evaluation, in order to use all the corpus for training and testing, 10-fold cross validation was used.

5.4 Results

In this section, we report the experiments undertaken and their results. First, we describe the results obtained by running the classifiers trained with no previous sampling. Next, the results obtained by applying the sampling algorithms separately are described. Finally, we describe the results obtained when sampling algorithms are combined.

We assess our results using the well-known F-measure and AUC. The main purpose of reporting results in F-measure is to allow for comparison with other results previously published in the area. However, the comparison should be made with a word of caution, as previously discussed, since the F-measure is influenced by the classes prior probabilities. AUC is the state-of-the-art measure for performance assessment of classifiers in presence of class-imbalanced data.

In the following tables, the results are displayed from left to right going from the least to the most imbalanced dataset. Hence, the results obtained with the Portuguese dataset (PT) appear first, then those with the datasets for Dutch (DU) and finally with the one for English (EN).

5.4.1 No Sampling Algorithms

We start by showing the performance results of learned models when no sample algorithm is applied, in Table 5.2.

They reveal that the performance of classifiers, in terms of F-measure, worsens when the class skewness increases, as we expected since this measure is influenced by the degree of imbalance. The same behavior is not so clear if we observe the AUC metric.

If we look at specific learners, the best performance in terms of AUC is obtained by Naïve Bayes and VFI algorithms. In the literature on imbalanced datasets, it is assumed that a classifier should present an AUC of at least 0.75 to be considered reliable (Bradley, 1997). VFI achieves this value with the first two datasets, and it gets very close to it

	PT		DU		EN	
	F-m	AUC	F-m	AUC	F-m	AUC
3-NN	0.09	0.63	0.02	0.54	0.01	0.67
C4.5	0.01	0.50	0.01	0.50	0.00	0.50
RF	0.08	0.64	0.01	0.65	0.00	0.69
NB	0.23	0.77	0.04	0.70	0.01	0.72
SVM	0.00	0.50	0.01	0.50	0.00	0.50
VFI	0.22	0.75	0.06	0.75	0.01	0.73

Table 5.2: Results obtained with the original unbalanced dataset

with the English dataset. Naïve Bayes only overcomes this threshold of 0.75 with the less imbalanced dataset, while for the other two it scores 0.70 and 0.72, respectively.

5.4.2 Single Sampling Algorithms

In this section, we present the experimental results obtained with a single pass of sampling algorithms. First, we present the results of random sampling, for balancing the training dataset so that it ends up with the same amount of negative and positive examples.

	PT		DU		EN	
	F-m	AUC	F-m	AUC	F-m	AUC
3-NN	0.59	0.62	0.22	0.54	0.65	0.68
C4.5	0.45	0.67	0.20	0.67	0.65	0.71
RF	0.25	0.63	0.20	0.64	0.68	0.69
NB	0.70	0.78	0.65	0.73	0.74	0.78
SVM	0.71	0.72	0.59	0.66	0.67	0.69
VFI	0.72	0.75	0.69	0.75	0.67	0.73

Table 5.3: Results obtained by Random Over-sampling algorithm

Table 5.3 and Table 5.4, respectively, display the results for balanced datasets obtained with random over-sampling and random under-sampling. In both cases, the F-measure score highly improves with respect to no sampling. While with the original dataset the best result was of 0.23 for Portuguese, with both sampling algorithms, a value around 0.70 is obtained for all the datasets. Regarding the AUC metric, it improves only with classifiers that obtained the worst performance when no sampling was

	PT		DU		EN	
	F-m	AUC	F-m	AUC	F-m	AUC
3-NN	0.65	0.70	0.18	0.54	0.71	0.76
C4.5	0.60	0.64	0.23	0.65	0.66	0.65
RF	0.63	0.69	0.55	0.65	0.73	0.75
NB	0.70	0.74	0.62	0.71	0.73	0.76
SVM	0.70	0.69	0.62	0.67	0.67	0.68
VFI	0.71	0.75	0.71	0.74	0.67	0.71

Table 5.4: Results obtained by Random Under-sampling algorithm

used.

We turn now to the experiments with direct sampling techniques. From Table 5.5 to Table 5.8, the results obtained with direct under-sampling algorithms are presented.

Turning our attention to these tables, it is possible to note the variation in performance when different under-sampling algorithms are used. At one extreme, CNN (Table 5.6) obtains slightly worse results than those obtained with random under-sampling. At the other extreme, Tomek Links (Table 5.8) shows high performance in terms of both F-measure and AUC.

In the case of ENN (Table 5.5), the number of majority class items deleted was not enough to notably modify the degree of imbalance, so that the proportion of negative and positive examples remains the same as the original dataset, with no sampling. As mentioned in Section 5.3.2, this algorithm was developed for cleaning data, and even if it is applied iteratively to the original dataset, eventually it will reach a stop point where no more examples can be deleted. As a consequence, the results are very similar to those obtained when no sampling algorithm is applied. We can conclude that for this kind of task (definition learning), in the way we have set the problem, this algorithm is not useful at all, at least when used alone.

	PT		DU		EN	
	F-m	AUC	F-m	AUC	F-m	AUC
3-NN	0.10	0.63	0.02	0.54	0.00	0.67
C4.5	0.03	0.51	0.00	0.50	0.00	0.50
RF	0.12	0.62	0.00	0.63	0.00	0.60
NB	0.25	0.78	0.05	0.71	0.00	0.72
SVM	0.00	0.50	0.00	0.50	0.00	0.50
VFI	0.22	0.74	0.06	0.75	0.07	0.73

Table 5.5: Results obtained by Edited Nearest Neighbor (ENN) algorithm

CNN presents a different behavior (Table 5.6). In terms of F-measure, it presents higher scores than those for the original dataset (i.e. no sampling) and quite similar to those of random sampling. In terms of AUC, the scores worsen. In particular, this deterioration is more evident with classifiers that are getting better results with the original dataset. For example, looking at the Portuguese experiment, we can see that the VFI obtained a score of 0.75 (Table 5.2) with the original dataset and with CNN, it

	PT		DU		EN	
	F-m	AUC	F-m	AUC	F-m	AUC
3-NN	0.54	0.54	0.61	0.58	0.60	0.55
C4.5	0.54	0.55	0.61	0.62	0.68	0.56
RF	0.58	0.62	0.63	0.66	0.60	0.51
NB	0.68	0.70	0.38	0.60	0.62	0.56
SVM	0.64	0.59	0.58	0.57	0.63	0.59
VFI	0.68	0.65	0.62	0.67	0.66	0.60

Table 5.6: Results obtained by Condensed Nearest Neighbor (CNN) algorithm

got 0.65. SVM obtained a score of 0.50 with the original dataset, while CNN now gets 0.59.

	PT		DU		EN	
	F-m	AUC	F-m	AUC	F-m	AUC
3-NN	0.41	0.75	0.27	0.70	0.20	0.68
C4.5	0.41	0.80	0.28	0.73	0.00	0.50
RF	0.22	0.82	0.24	0.75	0.22	0.82
NB	0.45	0.85	0.10	0.75	0.09	0.79
SVM	0.16	0.54	0.00	0.50	0.00	0.50
VFI	0.35	0.82	0.17	0.82	0.21	0.80

Table 5.7: Results obtained by Neighborhood Cleaning algorithm

NCL shows a significant improvement compared to random under-sampling (Table 5.7). Like ENN, this algorithm was not able to balance the dataset, but unlike ENN, it was able to modify the degree of imbalance. For each dataset, the following proportions were obtained: 1 to 3 for Portuguese (instead of 1 to 11), 1 to 17 for Dutch (instead of 1 to 42) and 1 to 28 for English (instead of 1 to 64). Since NCL did not achieve balanced datasets, it could be unfair to compare its results with the ones obtained with Tomek Links. For this reason, a second experiment was run, setting Tomek Links to reproduce the same proportion obtained by NCL for each dataset.

Table 5.9 shows the results of this experiment. In general, even when forced to reproduce the same data ratio returned by NCL, Tomek Links outperforms NCL, with some differences among the datasets. The advantage of Tomek Links is more evident with a less imbalanced dataset. As the imbalance increases, the two algorithms perform

	PT		DU		EN	
	F-m	AUC	F-m	AUC	F-m	AUC
3-NN	0.78	0.78	0.75	0.74	0.71	0.75
C4.5	0.85	0.89	0.81	0.87	0.82	0.78
RF	0.86	0.92	0.84	0.91	0.84	0.93
NB	0.86	0.89	0.86	0.90	0.63	0.83
SVM	0.82	0.81	0.88	0.88	0.73	0.74
VFI	0.79	0.87	0.85	0.92	0.68	0.84

Table 5.8: Results obtained by Tomek Links algorithm

	PT		DU		EN	
	F-m	AUC	F-m	AUC	F-m	AUC
3-NN	0.66	0.85	0.30	0.75	0.37	0.81
C4.5	0.68	0.83	0.32	0.69	0.00	0.50
RF	0.66	0.86	0.32	0.77	0.25	0.81
NB	0.71	0.90	0.10	0.76	0.04	0.80
SVM	0.44	0.64	0.00	0.50	0.00	0.50
VFI	0.51	0.84	0.16	0.82	0.15	0.79

Table 5.9: Results obtained by Tomek Links algorithm, when the dataset is not completely balanced

very similarly.

It is also interesting to compare these results obtained by non-balanced Tomek Links in Table 5.9 with those obtained by the same algorithm, only forced to return a perfectly balanced dataset (Table 5.8). In general, the balanced dataset obtains better results in terms of both F-measure and AUC, but the improvement is higher for the first metric. The only exception is the 3-NN based classifier. In this case, for all three datasets, the AUC value is higher when a non-perfectly balanced dataset is used.

Given these considerations, Tomek Links results to be the best under-sampling algorithm for the definition extraction task. In terms of the AUC metric, only in two cases, this algorithm did not reach the threshold of 0.75, namely with 3-NN when using the Dutch dataset and with SVM when using the English dataset. However, in both cases the AUC value is 0.74, very near to the threshold. The best classifier, Random Forest, outperforms all other classification algorithms for all datasets, with AUC scores above 0.90. Voted Feature Interval and Naïve Bayes algorithms follow at a short distance.

	PT		DU		EN	
	F-m	AUC	F-m	AUC	F-m	AUC
3-NN	0.61	0.74	0.63	0.73	0.70	0.75
C4.5	0.79	0.88	0.89	0.95	0.81	0.81
RF	0.77	0.92	0.86	0.98	0.71	0.85
NB	0.70	0.78	0.67	0.93	0.79	0.86
SVM	0.72	0.72	0.68	0.72	0.69	0.70
VFI	0.69	0.87	0.67	0.87	0.65	0.79

Table 5.10: Results obtained by SMOTE algorithm

Regarding oversampling, when SMOTE (Table 5.10) is applied, results are very similar to those obtained with Tomek Links, but with a slight difference. The threshold of 0.75 for AUC is not reached in five settings, namely for all datasets when SVM is used, and for Portuguese and Dutch datasets with 3-NN based classifiers. When looking at specific classifiers, Random Forest maintains its advantage, but this time it is followed by C4.5 and then by Voted Feature Interval and Naïve Bayes.

Finally, it is interesting to note that for less imbalanced datasets, Tomek Links outperforms SMOTE. When the datasets are more imbalanced, the two algorithms deliver similar performances.

5.4.3 Combining Sampling Algorithms

In this section, we present and discuss the results obtained by combining under- and over-sampling algorithms.

We opted for not reporting the results of the experiments carried out when over-sampling is applied first. This is so because, as the number of examples increase, some algorithms are computationally very expensive, but above all because the performance was slightly worse than the one obtained by doing under-sampling first.

Each experiment is repeated for under-sampling the majority class items to 75%, 50% and 25% of its initial size.

In a first experiment, we combined random under-sampling with a random over-sampling algorithm first and then with SMOTE. Table 5.11 shows the performance of classifiers for these two combinations.

When combining the two random algorithms, the performance does not improve, and the scores are quite similar to those obtained using just one such sampling algorithm.

	Random		SMOTE	
	F-m	AUC	F-m	AUC
	75%			
3-NN	0.67	0.71	0.66	0.74
C4.5	0.52	0.61	0.62	0.71
RF	0.49	0.67	0.67	0.78
NB	0.62	0.71	0.78	0.88
SVM	0.67	0.69	0.72	0.72
VFI	0.72	0.76	0.70	0.82
	50%			
3-NN	0.70	0.70	0.65	0.75
C4.5	0.47	0.63	0.76	0.85
RF	0.32	0.63	0.77	0.88
NB	0.68	0.77	0.88	0.95
SVM	0.71	0.72	0.73	0.73
VFI	0.72	0.76	0.69	0.86
	25%			
3-NN	0.61	0.64	0.63	0.74
C4.5	0.45	0.68	0.81	0.88
RF	0.28	0.65	0.80	0.91
NB	0.69	0.78	0.92	0.96
SVM	0.72	0.73	0.73	0.72
VFI	0.71	0.75	0.70	0.87

(a) PORTUGUESE

	Random		SMOTE	
	F-m	AUC	F-m	AUC
3-NN	0.47	0.57	0.64	0.70
C4.5	0.50	0.68	0.82	0.89
RF	0.18	0.68	0.81	0.94
NB	0.67	0.74	0.95	0.97
SVM	0.63	0.66	0.71	0.72
VFI	0.70	0.73	0.67	0.85
3-NN	0.30	0.54	0.65	0.73
C4.5	0.30	0.66	0.82	0.91
RF	0.07	0.66	0.84	0.97
NB	0.65	0.74	0.98	0.98
SVM	0.61	0.66	0.72	0.73
VFI	0.69	0.75	0.67	0.87
3-NN	0.21	0.53	0.64	0.73
C4.5	0.23	0.65	0.85	0.93
RF	0.02	0.67	0.85	0.98
NB	0.66	0.74	0.98	0.99
SVM	0.61	0.67	0.69	0.72
VFI	0.69	0.75	0.67	0.87

(b) DUTCH

	Random		SMOTE	
	F-m	AUC	F-m	AUC
3-NN	0.66	0.70	0.72	0.79
C4.5	0.68	0.73	0.78	0.77
RF	0.66	0.69	0.76	0.85
NB	0.72	0.79	0.81	0.89
SVM	0.67	0.68	0.69	0.69
VFI	0.69	0.73	0.64	0.76
3-NN	0.67	0.70	0.73	0.78
C4.5	0.69	0.72	0.82	0.81
RF	0.66	0.70	0.79	0.87
NB	0.74	0.78	0.85	0.92
SVM	0.68	0.69	0.68	0.69
VFI	0.69	0.74	0.63	0.77
3-NN	0.64	0.66	0.71	0.76
C4.5	0.64	0.71	0.80	0.80
RF	0.64	0.68	0.77	0.86
NB	0.73	0.79	0.87	0.94
SVM	0.67	0.68	0.68	0.69
VFI	0.68	0.73	0.65	0.78

(c) ENGLISH

Table 5.11: Results obtained by Random Under-sampling algorithms in combination with Over-sampling algorithms (Random and SMOTE)

Something similar happens when random under-sampling is followed by SMOTE. Here, the results obtained are about the same as those obtained by SMOTE alone.

A different scenario can be observed when we turn to Table 5.12, which displays the results obtained by using Tomek Links as the first algorithm in the sequence of sampling algorithms.

As for the combination of Tomek Links with random over-sampling, only when the majority class items are reduced to 75%, results are better or similar to those obtained using Tomek Links alone.

As for the combination of Tomek Links with SMOTE, the situation is more complex. For 3-NN, SVM and VFI classifiers, the best results are obtained when Tomek Links

	Random		SMOTE	
	F-m	AUC	F-m	AUC
	75%			
3-NN	0.81	0.83	0.83	0.85
C4.5	0.73	0.81	0.80	0.85
RF	0.83	0.88	0.84	0.91
NB	0.84	0.90	0.88	0.95
SVM	0.82	0.81	0.83	0.82
VFI	0.75	0.86	0.76	0.90
	50%			
3-NN	0.77	0.77	0.79	0.80
C4.5	0.65	0.76	0.79	0.87
RF	0.68	0.80	0.84	0.91
NB	0.81	0.87	0.91	0.97
SVM	0.79	0.79	0.81	0.80
VFI	0.75	0.83	0.73	0.90
	25%			
3-NN	0.71	0.70	0.72	0.77
C4.5	0.52	0.70	0.76	0.87
RF	0.42	0.70	0.80	0.91
NB	0.76	0.83	0.92	0.97
SVM	0.74	0.74	0.78	0.77
VFI	0.75	0.79	0.71	0.89

(a) PORTUGUESE

	Random		SMOTE	
	F-m	AUC	F-m	AUC
3-NN	0.73	0.76	0.77	0.79
C4.5	0.69	0.78	0.87	0.91
RF	0.64	0.82	0.88	0.96
NB	0.78	0.85	0.95	0.98
SVM	0.81	0.82	0.84	0.85
VFI	0.74	0.84	0.74	0.89
3-NN	0.66	0.70	0.78	0.80
C4.5	0.53	0.79	0.84	0.91
RF	0.37	0.77	0.89	0.98
NB	0.74	0.80	0.98	0.99
SVM	0.72	0.75	0.80	0.82
VFI	0.73	0.81	0.70	0.89
3-NN	0.48	0.61	0.72	0.76
C4.5	0.31	0.69	0.86	0.94
RF	0.16	0.70	0.85	0.98
NB	0.69	0.76	0.98	0.99
SVM	0.65	0.70	0.74	0.76
VFI	0.70	0.77	0.68	0.88

(b) DUTCH

	Random		SMOTE	
	F-m	AUC	F-m	AUC
3-NN	0.79	0.85	0.80	0.84
C4.5	0.74	0.81	0.82	0.83
RF	0.79	0.85	0.82	0.90
NB	0.84	0.89	0.85	0.93
SVM	0.76	0.77	0.81	0.82
VFI	0.74	0.81	0.64	0.82
3-NN	0.74	0.76	0.79	0.80
C4.5	0.70	0.83	0.85	0.86
RF	0.74	0.78	0.79	0.90
NB	0.81	0.87	0.90	0.96
SVM	0.78	0.78	0.80	0.80
VFI	0.75	0.80	0.66	0.82
3-NN	0.70	0.73	0.73	0.79
C4.5	0.71	0.80	0.85	0.86
RF	0.68	0.74	0.79	0.88
NB	0.78	0.83	0.91	0.96
SVM	0.71	0.73	0.74	0.75
VFI	0.72	0.78	0.66	0.81

(c) ENGLISH

Table 5.12: Results obtained by Tomek Links in combination with Over-sampling algorithms (Random and SMOTE)

reduce the dataset to 75%. Random Forest and Voted Feature Intervals work better when the reduction is 50%. And finally C4.5 achieves the best performance with the third setting, with the maximum reduction of imbalance, down to 25%.

It is interesting to note that in the case of the Portuguese dataset, the least imbalanced one, the improvement for most classifiers is not very significant, except when Naïve Bayes is used. But as the skewness increases, the combination of the two algorithms generate better results in terms of F-measure and AUC for almost all classifiers when comparing the results obtained with either Tomek or SMOTE alone. For the Dutch and English datasets, this improvement occurs not only with the best setting, but with all three reduction levels considered.

Finally, it is worth noting that, for all classifiers in all datasets, the threshold of 0.75 of AUC is achieved.

	Random		SMOTE	
	F-m	AUC	F-m	AUC
	75%			
3-NN	0.75	0.73	0.67	0.73
C4.5	0.66	0.74	0.69	0.81
RF	0.64	0.76	0.75	0.87
NB	0.78	0.84	0.84	0.95
SVM	0.77	0.73	0.72	0.76
VFI	0.73	0.82	0.64	0.89
	50%			
3-NN	0.74	0.7	0.71	0.74
C4.5	0.52	0.71	0.76	0.87
RF	0.57	0.72	0.79	0.88
NB	0.78	0.83	0.88	0.96
SVM	0.75	0.73	0.76	0.76
VFI	0.73	0.82	0.69	0.89
	25%			
3-NN	0.70	0.68	0.70	0.74
C4.5	0.56	0.73	0.78	0.88
RF	0.49	0.70	0.79	0.90
NB	0.75	0.83	0.91	0.97
SVM	0.73	0.73	0.75	0.75
VFI	0.73	0.79	0.70	0.90

(a) PORTUGUESE

	Random		SMOTE	
	F-m	AUC	F-m	AUC
3-NN	0.66	0.68	0.79	0.81
C4.5	0.52	0.76	0.88	0.94
RF	0.76	0.76	0.81	0.82
NB	0.76	0.81	0.97	0.99
SVM	0.43	0.79	0.90	0.97
VFI	0.73	0.80	0.75	0.89
3-NN	0.59	0.66	0.74	0.78
C4.5	0.52	0.76	0.88	0.94
RF	0.70	0.73	0.78	0.79
NB	0.74	0.79	0.98	0.99
SVM	0.30	0.73	0.87	0.97
VFI	0.71	0.78	0.69	0.88
3-NN	0.31	0.51	0.70	0.75
C4.5	0.36	0.70	0.87	0.94
RF	0.64	0.70	0.74	0.76
NB	0.69	0.76	0.98	0.99
SVM	0.10	0.68	0.85	0.98
VFI	0.71	0.77	0.69	0.88

(b) DUTCH

	Random		SMOTE	
	F-m	AUC	F-m	AUC
3-NN	0.80	0.84	0.80	0.81
C4.5	0.74	0.82	0.74	0.85
RF	0.82	0.88	0.87	0.92
NB	0.80	0.89	0.91	0.96
SVM	0.81	0.80	0.82	0.80
VFI	0.77	0.80	0.69	0.82
3-NN	0.74	0.78	0.74	0.78
C4.5	0.71	0.81	0.79	0.86
RF	0.74	0.83	0.83	0.90
NB	0.80	0.89	0.90	0.96
SVM	0.77	0.76	0.76	0.75
VFI	0.75	0.80	0.65	0.81
3-NN	0.72	0.75	0.73	0.77
C4.5	0.69	0.77	0.80	0.83
RF	0.70	0.75	0.82	0.90
NB	0.78	0.84	0.90	0.95
SVM	0.69	0.71	0.72	0.74
VFI	0.70	0.77	0.67	0.81

(c) ENGLISH

Table 5.13: Results obtained by Neighborhood Cleaning Rule (NCL) algorithm in combination with Over-sampling algorithms (Random and SMOTE)

When we turn to NCL, in Table 5.13, a similar scenario is observed. As the imbalance increases, the best results are obtained with the combination of the two sampling algorithms. In this case, for the Dutch and English datasets, the best performance is when NCL reduces the initial datasets to 75% of their size. For English, with this specific setting, all results are better than when SMOTE is used alone. On the other hand, for the Portuguese and Dutch datasets, it is possible to note that AUC gets worse with 3-NN, C4.5 and RF classifiers and improves with the other three.

Given the specificity of the remaining two under-sampling algorithms, ENN and

	Random		SMOTE	
	F-m	AUC	F-m	AUC
3-NN	0.58	0.63	0.63	0.75
C4.5	0.45	0.68	0.80	0.89
RF	0.26	0.65	0.80	0.93
NB	0.69	0.78	0.93	0.97
SVM	0.71	0.72	0.72	0.73
VFI	0.72	0.76	0.68	0.87

(a) PORTUGUESE

	Random		SMOTE	
	F-m	AUC	F-m	AUC
3-NN	0.22	0.54	0.62	0.73
C4.5	0.19	0.67	0.89	0.95
RF	0.02	0.64	0.86	0.99
NB	0.64	0.73	0.99	0.99
SVM	0.60	0.67	0.68	0.72
VFI	0.69	0.75	0.66	0.87

(b) DUTCH

	Random		SMOTE	
	F-m	AUC	F-m	AUC
3-NN	0.65	0.68	0.70	0.75
C4.5	0.66	0.71	0.81	0.82
RF	0.64	0.68	0.71	0.85
NB	0.74	0.78	0.87	0.95
SVM	0.68	0.69	0.69	0.71
VFI	0.67	0.73	0.65	0.79

(c) ENGLISH

Table 5.14: Results obtained by Edited Nearest Neighbor Rule (ENN) algorithm in combination with Over-sampling algorithms (Random and SMOTE)

CNN (see subsection 5.3.2), it is not possible to execute the three variants (75%, 50% and 25%) of each experiment.

The combination of ENN with over-sampling, whose results are displayed in Table 5.14, permit to achieve some improvement. By comparing these results with those in Table 5.10, for SMOTE alone, we observe that there is a small improvement for most classifiers when the combination of the two algorithms is used.

Finally, Table 5.15 displays the results for the under-sampling with CNN. For the first time, it is possible to find results improving with some datasets but not with others.

In this experiment, for the Dutch dataset, the results obtained in terms of F-measure are comparable to the ones obtained with SMOTE alone. As for the AUC scores, these are just slightly worse than those obtained with SMOTE alone.

With the Portuguese and English datasets, in turn, results are worse than the results obtained with either CNN alone, or with SMOTE alone.

To understand these results, it is important to notice that, for different datasets, CNN returns different proportions between the positive and the negative classes. In particular, in this experience, it delivered 1:2 for the Portuguese dataset, 1:7 for the Dutch dataset, and 1:1.5 for the English one. Given the lowest ratio obtained for the Dutch dataset, this implied that the larger portion of the balancing work was left to the SMOTE algorithm, and this explains why the final results were better in this case.

	Random		SMOTE	
	F-m	AUC	F-m	AUC
3-NN	0.52	0.59	0.51	0.66
C4.5	0.43	0.54	0.56	0.65
RF	0.35	0.59	0.52	0.72
NB	0.55	0.52	0.62	0.69
SVM	0.68	0.64	0.60	0.62
VFI	0.69	0.69	0.61	0.77

(a) PORTUGUESE

	Random		SMOTE	
	F-m	AUC	F-m	AUC
3-NN	0.49	0.53	0.62	0.65
C4.5	0.28	0.57	0.78	0.85
RF	0.08	0.63	0.83	0.93
NB	0.62	0.69	0.92	0.95
SVM	0.64	0.65	0.66	0.67
VFI	0.70	0.67	0.66	0.82

(b) DUTCH

	Random		SMOTE	
	F-m	AUC	F-m	AUC
3-NN	0.62	0.49	0.56	0.50
C4.5	0.58	0.59	0.59	0.54
RF	0.58	0.53	0.59	0.61
NB	0.00	0.50	0.61	0.50
SVM	0.61	0.56	0.62	0.50
VFI	0.62	0.55	0.59	0.66

(c) ENGLISH

Table 5.15: Results obtained by Condensed Nearest Neighbor Rule (CNN) algorithm in combination with Over-sampling algorithms (Random and SMOTE)

	Rule Based			10-fold			Testing Corpus		
	P	R	F-m	P	R	F-m	P	R	F-m
PT	0.32	0.66	0.43	0.95	0.90	0.92	0.78	0.86	0.82
EN	-	-	-	0.92	0.90	0.91	0.81	0.89	0.84
DU	-	-	-	0.98	0.96	0.98	0.97	0.91	0.94

Table 5.16: Results using different strategies with copula definitions

5.4.4 Summary

In order to conclude, we can observe that the most effective setup was a combination of Tomek Links followed by SMOTE, as sampling algorithms, with Naïve Bayes as learning algorithm. This combination was particularly effective in dealing with datasets with higher imbalances. Now, we want to test our strategy by means of a different evaluation method. Results presented in this Chapter were obtained using 10-fold cross validation, as we have few positive examples. We also split each corpus in two sub-corpora, a development corpus, consisting of 75% of the whole corpus used for training the classifier and an held out 25% of the corpus, used to test the classifier. Regarding the Portuguese corpus, the division in two sub-corpora follows the division carried out in the development of the rule-based system.

In Table 5.16, we present the best results obtained with the three different approaches, that is rule-based, machine learning with 10-fold cross validation and machine learning with a separate testing corpus.

In general, it is possible to observe that there is a certain worsening in the results

obtained with the testing corpus in comparison with the 10-fold cross validation strategy. This can be due to the fact that the 10-fold cross validation uses 90% of the dataset for training the classifier and the remaining 10% of the for testing it, while in the experiment carried out splitting the corpus in two parts, only the 75% of the corpus were used for training the classifier. This means that the number of positive examples in this last setup was smaller in comparison with the 10-fold cross validation, and probably the resulting amount of example is not enough to train a good classifier. In order to verify this hypothesis, we carried out the same experiments modifying the size of the training and the testing corpora, in this way the training was composed by the 90% of the corpus and the remaining 10% was used for the testing phase. For all the languages, we obtained results very similar to the ones obtained with the 10-fold cross validation, with a difference of no more than 2 points.

Regarding Portuguese, results obtained with the rule-based module can be directly compared with the ones obtained with the machine learning approach with the training corpus, as the same corpora for training and for testing were used. The machine learning approach largely overcomes the baseline constituted by the rule-based approach.

5.5 Discussion

From the systematic experimentation carried out with respect to the task of definition extraction and reported above, a number of lessons can be gathered.

Sampling imbalanced datasets frequently improves the performance of classifiers over the baseline, which is in the range of 0.73-0.77 in terms of AUC score.

However, not all sampling techniques are equally suited to foster this improvement. Some of them add very little improvement, if any. That is the case of random over- and under-sampling, and ENN. Some others may even deteriorate the results to deliver scores clearly below the baseline. That is the case of CNN.

For under-sampling, NCL and Tomek Links consistently helped to improve the performance of classifiers, as well as SMOTE, for over-sampling, with AUC results in the range of 0.82-0.98. Tomek Links should be pointed out as one of the best options as it exhibits an equal top-performance for datasets with different imbalance degrees, with AUC in the range of 0.92-0.93.

In general, combining under-sampling with over-sampling provides better results than when only one of them is used. Better results were obtained performing under-sampling before over-sampling, and using direct over-sampling with SMOTE instead of just plain random over-sampling. The exception is found again when CNN is chosen as under-sampling method. But all other combinations (including those with random under-sampling as the first step) have consistently shown to be able to raise the AUC scores to the range of 0.94-0.99.

The best performing combinations are the ones that result from combining the algorithms that had been shown to be the best in their categories when applied in isolation. That is, the best results are obtained with NCL or Tomek Links for under-sampling and SMOTE for over-sampling.

Which sampling method should be preferred will largely depend on the computational cost associated with the use of these two algorithms. In fact, as mentioned above, the use of over-sampling algorithms increases the computational cost because it increases the number of examples that will be used to build the classifier. In contrast, Tomek Links is a very complex algorithm that, for every example, considers all examples in order to identify a link. This means that the time needed to under-sample the dataset is polynomially related to the size of the original dataset. Due to the growth in computational capacity of computers, it is likely that this question may not represent a real issue for many applications.

In general, there seems to exist a tendency for the largest extension of the under-sampling to permit the best results, and hence for less catch-up work to be required for the over-sampling step. When under-sampling to just 75%, the scores are in the range of 0.93-0.99, while they are in the range of 0.96-0.99 for under-sampling to as much as 25%. But more firm conclusions on this respect would perhaps need to be based on datasets with a larger range of number of tokens.

Interestingly, the best results – that is, when under-sampling and over-sampling are combined – are consistently obtained by the Naïve Bayes classifier. Also consistently, the second best option is the Random Forest classifiers, though at a clear distance behind.

It is interesting to note that these two classifiers are among the best ones even when no sampling algorithms are used. This result suggests that, for the learning algorithms used in this work, Naïve Bayes and Random Forest have the most suited learning bias

for the task of definition extraction. Therefore, the sampling algorithms are responsible for *just* leveraging off the classification performance of these algorithms, given the imbalanced characteristic of the data. The reason of why the learning bias of these algorithms is suited for definition extraction is out of the scope of this work. However, we note that it is in conformity with theoretical and practical results present in the literature. For instance, Naïve Bayes frequently presents good performance in Natural Language Processing (Roth, 1999) even when the model assumptions of independence does not hold (Zhang, 2005). Regarding Random Forest, ensemble classifiers are one of the most effective computationally intensive procedures to improve on unstable estimators or classifiers, being useful especially for high dimensional data set problems (Biau, 2012).

The same tendencies can be observed with F-measure scores. The best combinations of under-sampling, over-sampling and classifier algorithms consistently deliver performances scoring in the range of 0.87-0.99 in terms of F-measure.

Looking more closely at the scores obtained by Naïve Bayes and Random Forest, there is a difference in terms of precision and recall. For all the three languages, Random Forest classifiers were able to obtain a recall close to 1, that is all the definitions were correctly identified, but there is a larger number of non-definitions classified as positive examples in comparison to what happens with Naïve Bayes classifiers.

When comparing with previous work on definition extraction, our results outperform all the systems that have used learning algorithms, confirming the importance of sampling techniques in supporting the definition extraction task.

Westerhout & Monachesi (2007), using the same corpus we used for Dutch, report an F-measure of 0.73, obtained with a combination of syntactic rules and a Naïve Bayes classifier for Dutch. Przepiórkowski *et al.* (2008), in turn, with a similar approach, but for the Polish language, obtained an F-measure of 0.35. Additionally, it is very important to note that, while our experiments just use bigrams of POS tags as features, all these previous works use a combination of sophisticated, and highly language-dependent features in order to reach the best results.

As in the last years Balanced Random Forest has been successfully used in different classification tasks (see for instance (Acedański *et al.*, 2012)) and in order to try a more direct comparison between our results and the ones obtained by (Degórski *et al.*, 2008a) and (Westerhout, 2009), we run this algorithm on our datasets. In this way we obtained

an F-measure of 0.73, 0.58 and 0.48 for Portuguese, Dutch and English, respectively. With respect to results obtained with different sampling techniques presented here, these scores are comparable to the ones returned by random sampling. With respect to (Westerhout, 2009), where the same Dutch dataset was used, resulting in an F-measure of 0.78, a direct comparison is not possible, since Balanced Random Forest was used as a filtering module after the application of a quite elaborate pattern module, and also because more sophisticated features were used. In the case of (Degórski *et al.*, 2008a), which reports an F-measure of 0.32, a different dataset and feature selection were used, making direct comparison not viable.

If we turn now to systems based only on pattern-matching ensured by hand-crafted rules, the state of the art in the area is represented by systems such as DEFINDER (Klavans & Muresan, 2001), which is reported to have an F-mesure of 0.80. Though not strictly comparable due to the use of different experimental conditions, including different datasets for evaluation, our approach seems to deliver results above this performance by a large margin, with scores in the range of 0.90-0.99.

Also when put into contrast with the usage of this same approach to other natural language processing tasks, our results seem to be very competitive. As discussed in Section 5.2.2, Liu *et al.* (2006) applied a combination of under- and over sampling to sentence boundary detection in speech, showing that under-sampling and SMOTE offer the best results with an AUC of 0.89 (the baseline being 0.80). However, they did not experiment intelligent under-sampling methods such as Tomek Links.

In another task, of automated annotation of keywords, Batista *et al.* (2003) gets the best results in terms of AUC with an improvement of 4 percentage points on the original dataset using a combination of SMOTE with Tomek Links.

In our case, with scores in the range of 0.94-0.99, the improvement with regards to the baseline of 0.73-0.77, is at least between 17 and 22 percentage points, demonstrating how these methods can be effective for our definition extraction application.

Finally, it is worth noting that our results are in line with those reported in the literature on imbalanced datasets in general. In a comprehensive study on the behavior of several methods for balancing training data, using 11 UCI datasets,[1] Batista *et al.* (2004) showed that in most cases and with several datasets in different domains, SMOTE obtains the best performance. In general, they lead to a rise in the AUC metric of few

[1] http://archive.ics.uci.edu/ml/

percentage points (1 to 4) when the baseline was already high (more than 0.65), while when the baseline was under this value the improvement was comparable to the one obtained in our work, which was up to 34 percentage points.

5.5.1 Error Analysis

We also conducted a qualitative analysis on the examples not correctly classified by our approach. In particular, we analyzed the results of the best settings, that is when Tomek Links is paired with SMOTE and Naïve Bayes and Random Forest are used as classification algorithms.

As for all the three languages, Random Forest classifier was able to obtain a recall of 1, we do not have false negatives to analyze. Regarding false positive examples, that is, those sentences that were incorrectly classified as definitions, in a few cases we verified that they were good definitions but the humans annotators had just missed them. In about one fifth of the cases, sentences contained some definitional information, but the human annotators did not annotat these sentences as definitions because the definition spanned over several sentences. For instance, there are several cases where the defined term appears in a sentence, and its definition in the following sentence.

There is another set of sentences starting with demonstrative pronouns that are considered by the classifier as good definitions though they are not. Most of these sentences are referring to illustrations in the text. In this case, the results could be filtered either by improving the features that include information appearing before the definitional verb or with a simple grammar to be applied after or before the classifier.

In terms of false positive examples, there is no big difference between Naïve Bayes and Random Forest classifiers. Regarding false negative examples, they appear only in Naïve Bayes classifiers. They occur mostly when the definition is composed by only the *genus* or the *diferentiae*.

5.6 Verbal Definitions

Following the same method of the previous Chapter, the machine learning approach was applied also to other verbs definitions, in order to create a new baseline for this kind of definitions.

As for the pattern-based module, also in this case, we have followed a slightly different strategy for characterizing the definitions, focusing on the verbal pattern introducing the definition, that is the connector. This way, lemmas of the verbs that characterize definitions were included in the features list.

Experiments were conducted using only English and Portuguese corpora. Table 5.17 shows the degree of imbalance of the new datasets for other verbs definitions.

Only the results for the best configurations are presented, that is, when Tomek Links and SMOTE are used as sampling algorithms and the Naïve Bayes, Random Forest and VFI are used as learning algorithms.

	Sub Corpus		
	Sentences	Definitions	Ratio
English	3,640	124	1:28
Portuguese	744	98	1:7

Table 5.17: Dataset description for verbal definitions

Table 5.18 presents the performance of the classifiers when no sampling is applied to the datasets.

In terms of both F-measure and AUC, these results are worse than those obtained for copula definitions. In particular the AUC value never reaches the threshold of 0.75, regarded as the bottom line for considering a classifier reliable.

The performance improves when sampling algorithms are used (Tables 5.19 and 5.20). If we consider AUC values, the best performance is obtained when SMOTE is applied, with a value always equal or larger than 0.80 (with the only exception of VFI classifier applied to the English corpus). Instead, regarding F-measure, the best performance is obtained when Tomek Links algorithm is used.

	PT				EN			
	P	R	F-m	AUC	P	R	F-m	AUC
RF	0.24	0.09	0.13	0.60	0.25	0.03	0.05	0.51
NB	0.39	0.05	0.09	0.64	0.70	0.06	0.11	0.61
VFI	0.17	0.49	0.25	0.65	0.72	0.11	0.18	0.59

Table 5.18: Results obtaining with the original unbalanced dataset

	PT				EN			
	P	R	F-M	AUC	P	R	F-M	AUC
RF	0.88	0.52	0.65	0.82	0.88	0.53	0.67	0.81
NB	0.58	0.95	0.72	0.89	1.00	0.42	0.59	0.86
VFI	0.51	0.98	0.67	0.8	0.75	0.15	0.24	0.58

Table 5.19: Results obtained by SMOTE algorithm

	PT				EN			
	P	R	F-M	AUC	P	R	F-M	AUC
Random Forest	0.72	0.69	0.71	0.76	0.58	0.77	0.66	0.71
Naive Bayse	0.68	0.71	0.69	0.78	0.50	0.91	0.65	0.50
VFI	0.56	0.99	0.72	0.80	0.57	0.60	0.58	0.64

Table 5.20: Results obtained by Tomek Links algorithm

The combination of the two sampling algorithms positively influences F-measure, while AUC remains substantially unchanged (Table 5.21).

In general, the performance of classifiers handling other verbs definitions is worse than that for copula definitions. In this last case we got a range in the AUC scores between 0.94 and 0.99, and F-measure between 0.87 and 0.99. While for other verbs definitions, the best AUC is 0.89 and the best F-measure value is 0.79.

If these results are compared with those obtained by the rule-based approach, the improvement is considerable. The F-measure increases from 0.23 to 0.77. In terms of precision and recall, it is interesting to note that differently from other works using machine learning (Westerhout & Monachesi, 2008), recall does not get worse, but gets better. Precision benefits greatly from this approach, reaching a value of 0.77, while with the rule-based module it was 0.14.

In terms of state of the art, our results are already comparable or even better than other approaches, even if it is quite hard to compare results, as there are no other studies that focus on this specific definition type using machine learning.

For example, Borg *et al.* (2009), using genetic algorithms and the same English corpus we used, reports a precision of 0.62, a recall of 0.50 and a resulting F-measure of 0.57, without a definitions type distinction.

When comparing with copula definitions, clearly we notice that there is still room for

	P	R	F-m	AUC
	75%			
RF	0.77	0.73	0.75	0.81
NB	0.71	0.83	0.77	0.79
VFI	0.54	0.98	0.69	0.78
	50%			
RF	0.76	0.62	0.69	0.79
NB	0.66	0.88	0.76	0.87
VFI	0.52	0.99	0.68	0.81
	25%			
RF	0.76	0.60	0.67	0.76
NB	0.63	0.93	0.75	0.89
VFI	0.51	0.98	0.67	0.79

(a) PORTUGUESE

P	R	F-m	AUC
0.81	0.76	0.78	0.85
0.71	0.87	0.78	0.85
0.53	0.14	0.22	0.47
0.75	0.79	0.77	0.85
0.98	0.49	0.65	0.83
0.67	0.08	0.13	0.42
0.76	0.52	0.62	0.8
0.99	0.65	0.78	0.87
0.66	0.11	0.19	0.53

(b) ENGLISH

Table 5.21: Results obtained by Tomek Links algorithm in combination with SMOTE algorithm

improvement. This relative poor performance is due to the fact that verbal definitions are very variable in their structure, and we need to feed machine learning algorithms with a great number of positive examples, where this variability is present. Therefore, it is not enough to delete non relevant examples or artificially increase the number of positives ones. In the next Chapter, we present a way to balance the dataset using positive examples automatically extracted from an external source.

5.7 Conclusions

The advances reported here result from a novel approach to the task of definition extraction. The major trend in the literature has been to build solutions for this task on the basis of some set of manually crafted patterns. In the present Chapter, we experimented thoroughly with an alternative solution based solely on machine learning techniques. The key twist to make such an approach not only viable but also with superior results was to focus on the issue of the imbalance of datasets. This permitted to take advantage of the solutions that have been put forward to this problem in recent years, and eventually find out that they allow for a notable breakthrough in terms of the task of definition extraction.

The results obtained with our experiments show that it is feasible to consistently bring the performance of an automatic extractor of definitions to score in the range of 0.95-0.99 in terms of AUC (and in the range of 0.90-0.99 in terms of F-measure). The improvement with regards to the baseline of 0.73-0.77 is thus at least between 17 and 22 percentage points.

Very interestingly, these advances were obtained not only by dispensing with manually crafted patterns, but also by resorting only to bigrams of POS tags (as features for the classifiers), thus greatly improving the transportability of the approach both across domains and languages.

On par with these overall advances, by systematically experimenting with different paradigms of learning algorithms in combination with different sampling techniques, and with datasets with different imbalance rates, it was possible to draw some finer conclusions regarding the best practice to adopt when handling automatic definition extraction.

In particular, the most effective setup was shown to be a combination of Tomek Links, for under-sampling, followed by SMOTE, for over-sampling, even more so when datasets with higher imbalances have to be dealt with.

As for the setup with only one step of dataset imbalance reduction, we can conclude that Tomek Links is the best choice. This algorithm improves the result for all classifiers and for all datasets, independently of the degree of their imbalance or of their language, while SMOTE tends to be less effective with higher data skewness.

As for the classifiers used for this task of definition extraction, Random Forest and Naïve Bayes present the best results in almost all the experiments, with a significant difference between the two classifiers in terms of recall and precision: Naïve Bayes performs better in terms of precision, while Random Forest get a higher score for recall.

In order to generalize our results, we tested our approach with verbal definitions. In this case, even if the degree of imbalance is slightly smaller than the imbalance in copula definitions, result are worse, presenting an F-measure around 0.77 and an AUC around 0.79, confirming the combination of Tomek Links and SMOTE as the best sampling method and Naïve Bayes as the best learning algorithm.

We can thus observe that, under this approach, the best way to construct a definition extractor is to build a Naïve Bayes classifier after sampling the dataset using a combination of Tomek Links followed by SMOTE.

As a final remark, it is worth noting that the present results not only represent a progress in the area of automatic extraction of definitions, but they also reinforce the value of using sampling techniques in the field of Natural Language Engineering, where most tasks and tools rely on datasets with notorious imbalance. The present research adds to the seminal work that point towards the important research avenue of seriously applying sampling techniques to mitigate the adverse bias induced by highly imbalanced datasets and thus greatly improving the performance of a range of tools for Natural Language Processing.

Chapter 6

Verbal Definitions: Wikipedia as a corpus

6.1 Introduction

This Chapter will present the strategy adopted to deal with verbal definitions. This is with the same approach as used for copula definitions, by resorting to machine learning algorithms, but with a different strategy regarding the sampling of the dataset.

As discussed in the previous Chapter, the use of sampling techniques has proved ineffective in achieving satisfactory results with this type of definitions. A possible explanation lies in the fact that, while copula definitions are introduced by the same verb and, for this reason, they have a more homogeneous structure, verbal definitions are introduced by a number of different verbs, ending up in a higher number of heterogeneous structures. The replication of positive examples or the elimination of confusing negative examples does not improve the performance of a classifier if the problem consists in a low proportion of positive examples, representing the different structures of definitions. Consequently, in order to train a good classifier, we need examples covering the multiplicity of different structures characterizing verbal definitions.

The solution proposed here is to increase the number of positive examples not by means of replicating them, but by finding new real definitions to be added to the dataset. As mentioned several times in this dissertation, corpora annotated with definitions are not easily available and annotating a new corpus is a long and costly process. The challenge is to come up with a source and a method to increase automatically the number of definitions, without the need for human annotators.

We propose to use the Wikipedia on-line encyclopedia[1] as a source of definitions, by exploiting its particular structure in order to automatically extract definitions to use to train a classifier. The convenience of using Wikipedia as a source for definitions is based on the peculiar structure of its articles, which follow well-defined rules stating that the first paragraph of each article should give a brief definition of the topic described in the article.

In this Chapter, we discuss how Wikipedia can be used as a starting point in the construction of a general corpus of definitions. By general corpus, we mean a corpus made of articles on general topics in different domains. This corpus may be also used to extract lexical information, such as as the verbal patterns characterizing verbal definitions.

In the last part of this Chapter, we present the experimental settings for building a classifier for verbal definitions for both the Portuguese and the English languages, using definitions gathered from Wikipedia. Based on the knowledge gained with the copula definitions in terms of learning algorithms, we just present the performance of the best algorithms and discuss the results.

6.2 Wikipedia

Wikipedia probably represents one of the largest open source language repositories, with more than 7.5 million articles in more than 250 different languages. Besides the value provided by its size, another advantage is to be found in the structure and metadata enriching the plain text. Articles in Wikipedia are not isolated pieces of information. They are linked to each other through both a great number of links and a structured category system.

Due to this rich structured information, Wikipedia has been considered as an invaluable resource for NLP and Data Mining tasks (Nakayama *et al.*, 2008), and has been used in a variety of NLP-related task, such as text classification (Tomuro & Shepitsen, 2009), information retrieval (Zesch & Gurevych, 2007), question answering, computing semantic relatedness (Zesch *et al.*, 2008), or textual entailment (Zanzotto & Pennacchiotti, 2010).

[1] www.wikipedia.com

Regarding definition extraction, Wikipedia has already been used as a source for definitions. Fahmi & Bouma (2006) collected a sub-corpus of the Dutch version of Wikipedia in the area of healthcare, searching for sentences where the verb *to be* was the main verb of the sentence. They manually annotated these sentences as being definitions or not. The resulting dataset was used to build different classifiers looking for the best feature configuration. Navigli & Velardi (2010) extracted 4,619 Wikipedia sentences, containing 1,908 definitional and 2,711 non-definitional sentences in different domains, that were manually identified. These authors used algorithms based on word-class lattices to classify definitions. They applied two different lattice-based algorithms to this dataset obtaining a precision of 0.99 and a recall of 0.61, with a resulting F-measure of 0.75.

Our approach is quite innovative, as we use Wikipedia in order to automatically extract verbal definitions, that are used to integrate an existing dataset of definitions, to support the construction of a classifier.

One of the issues to be addressed regards how to ensure that the corpus extracted from Wikipedia is a general balanced corpus, where no specific domain is over-represented and the articles are not about very specific concepts. For this reason, it is very important to consider how to select Wikipedia articles to be used as sources of definitions. For this specific propose, we exploit Wikipedia category structure.

6.2.1 Structure: Articles and Categories

Articles in Wikipedia are organized in a taxonomy-like structure by means of categories. These categories do not form a strict hierarchy or tree, since each article may appear in more than one category, and each category may appear in more than one parent category, allowing multiple categorization schemes to co-exist simultaneously. Each category may have an arbitrary number of subcategories, where a subcategory is typically established because a hyponymy or meronymy relation exists. For example, the category *Vehicle* has subcategories like *Aircraft* or *Watercraft*.[1]. Each category is linked to a non-predetermined number of articles that can vary from 0 to hundreds, such as the category *Comet*, which happens to be linked to over 200 articles describing different specific comets.[2]

[1] http://en.Wikipedia.org/wiki/Wikipedia:FAQ/Categorization
[2] http://en.wikipedia.org/wiki/Category:Comets

When browsing Wikipedia categories for articles there are two top categories, parents of all other categories, denoting a top-level place to start browsing the graph of the knowledge. They represent a top level entry in terms of encyclopedia article function and content. These two top categories are "*Fundamentals*"[1] and "*Main Topics*".[2]

Fundamentals is intended to contain all and only the few most *Fundamental* ontological categories which can reasonably be expected to contain every possible Wikipedia article under their category trees. This category has four subcategories: *Concepts*, *Life*, *Matter*, *Society*.

Main Topics is an alternative root category, based on a somewhat more detailed initial classification. It has twenty-two sub-categories: *Agriculture*, *Arts*, *Belief*, *Business*, *Chronology*, *Culture*, *Education*, *Environment*, *Geography*, *Health*, *History*, *Humanities*, *Language*, *Law*, *Life*, *Mathematics*, *Nature*, *People*, *Politics*, *Science*, *Society*, *Technology*.

Although Wikipedia is composed by millions of articles, that are built in a cooperative way, covering the most different knowledge fields, its articles present a well-determined structure. This is due to the fact that Wikipedia contributors are invited to follow well-defined rules when creating a new article. In particular, Wikipedia states that the first paragraph of each article should define the topic with a neutral point of view, without being overly specific. It should establish the context in which the topic is being considered by supplying the set of circumstances or facts that surround it. Guidelines to help improve Wikipedia state that redundancy must be kept to a minimum in the first sentence.

In this way, all Wikipedia articles begin with a declarative sentence giving a concise definition, telling the non-specialist reader what the subject of the article is. Regarding the structure of this first sentence, generally the title of the article is the grammatical subject of this first sentence and it is in boldface.

As pointed our by Navigli & Velardi (2010) the first sentence of Wikipedia entries is, in the large majority of cases, a definition of the page title. It is possible to exploit this structure by automatically marking the first sentence of each article as definition, where the term defined is the title of the article, and the first verb in inflected form is the connector verb. Differently by Navigli & Velardi (2010) who manually annotated the

[1] http://en.wikipedia.org/wiki/Category:Fundamental_categories
[2] http://en.wikipedia.org/wiki/Category:Main_topic_classifications

sentences with the *definiens* and the *definiendum*, we developed a grammar in order to annotate the first sentence of each articles with this information and to discard sentences that not could be considered definitions.

6.2.2 Accessing Wikipedia

We accessed and analyzed Wikipedia through Java Wikipedia Library (JWPL), an open-source, Java-based application programming interface that allows one to access all information contained in Wikipedia (Zesch *et al.*, 2008).

While the original structure of the Wikipedia database is optimized for searching articles by keywords, which is performed by millions of users, this API is designed for NLP research and supports a wider range of access paths, including iteration over all articles, a query syntax, as well as efficient access to information like links, categories, and redirects. Thus, JWPL operates on an optimized database that is created in a one-time effort from the database dumps available from the Wikipedia Foundation.

JWPL supports retrieval by keywords or via a query interface that allows for wildcard matches, as well as retrieving subsets of articles or categories depending on parameters, like the number of tokens in an article or the number of ingoing links. It also allows to iterate over articles, categories, redirects, and disambiguation pages. It provides access to the article text (with markup information or as plain text), the assigned categories, the ingoing and outgoing article links, as well as all redirects that link to this article. The API also allows one to deal with categories, retrieving parent and child categories, as well as siblings and all recursively collected descendants and accessing to the articles within a specific category.

The two versions of Wikipedia (English and Portugues) used in this work are based on a dump available at `http://dumps.wikimedia.org/backup-index.html`. The English dump is dated from the 3rd of August 2011, while the Portuguese one is dated from the 30th of May 2011.

In Table 6.1 the size of the two Wikipedias is shown. English Wikipedia is just over 7 times larger than the Portuguese one in terms of the number the articles and it has 6 times more categories.

	EN	PT
Pages	8,739,845	1,240,318
Disambiguation Pages	133,475	16,377
Redirect Pages	4,998,892	471,489
Categories	744,971	116,885

Table 6.1: Wikipedia dump in number

6.3 Extract a Representative Corpus of Definitions

As discussed in Chapter 2, definitions represent a sub-language characterized by a characteristic lexical and syntactic structure, which can be further circumscribed given a specific knowledge domain. To prevent that the classifier for verbal definitions from being influenced by examples from a specific domain it is important to use a corpus where different domains are as evenly represented as possible, and where, for each domain, concepts belonging to the highest level of the conceptual tree are included. This means that if we have 10 different domains in a corpus, we want to have the same number of articles, and consequently definitions, for each domain. Furthermore, for each domain, we want articles on fundamental concepts to be present, instead of articles on very specific concepts. For instance, if one of the domains included in the corpus is about Botany, we want to include general articles on plants, seeds, plant morphology, etc., and we want to avoid articles on very specific elements such as an article on microcachrys (a species of dioecious conifers belonging to the podocarp family).

When using Wikipedia to build such a general corpus for improving automatic definition extraction, the issue of representativeness arises, due to the fact that the growth of Wikipedia is not controlled, and a particular area could be more extensive than another and there is no way to know where it happens.

As explained in the previous Section, articles in Wikipedia are organized in order to follow a hierarchical structure, from more general to more specific topics. Following this structure, it is possible to extract articles on general topics, selecting the articles directly linked to these top level categories. It also true that Wikipedia does not guarantee that the various domains are equally covered and with the same granularity. This means that when going down along the category structure, some domains may very soon include very specific articles. Furthermore, there are categories that are linked to a big number

of articles while others are linked to few or no articles. For example, the category *Agriculture* is linked to 191 articles while the category *Art* has only 7 articles directly linked to it.

For these reasons, we need a method to collect articles in order to reduce the likeliness of incurring in this problem. After analyzing the categories structure, we came up with two different algorithms to collect articles, and we applied these algorithms to the two top categories described in the previous section, that is *Main Topics* and *Fundamentals*.

6.3.1 Algorithms for Extracting Wikipedia Articles

A first algorithm ($Alg1$) recursively collects the same number of articles for each category below the top category. In this way we want to ensure that each domain, represented by the children of top categories, has the same likelihood of being represented.

A second algorithm ($Alg2$), first gathers all the articles linked to the top category children and then randomly collect the articles till the desired number of articles is reached. The rationale here is to collect articles linked to the most general categories. As for the first algorithm, if the number of articles is lower than the corpus size, the operation is repeated with the categories in the next level of the tree.

Using these algorithms, we extracted five corpora with different sizes, containing respectively 1000, 10000, 25000, 50000 and 100000 articles. All the articles linked to the category *People* and its descendants were excluded from the corpora, as they do not describe a concept. In this way, we wanted to find out which top category was better to start from, either "*Fundamentals*" or "*Main Topics*", and which algorithm could ensure a more general corpus.

We automatically extracted the first sentence of each article, as it represents a definition and marked the defined term and the connector verb. If the connector was the verb *to be*, the definition was marked as a copula definition, otherwise the definition was marked as a verbal definition. In Table 6.2, we present the proportion of copula and verbal definitions for the Portuguese and the English Wikipedia, for each corpus size. For both languages, copula definitions are predominant, but for Portuguese this predominance is larger and increases with the size of the corpus, while for English, the ratio between copula and verbal definitions does not suffer a similar variation.

	PT		EN	
Corpus	copula	verb	copula	verb
1000	73%	27%	66%	34%
10000	79%	21%	66%	34%
25000	80%	20%	67%	33%
50000	82%	16%	68%	32%
100000	86%	14%	69%	31%

Table 6.2: Statistics for Wikipedia definitions for both languages

6.3.2 Statistics

In order to evaluate which algorithm returns a more representative corpus of definitions, two different heuristics were used, based on the analysis of copula definitions. Several authors point out that words such as *technique*, *method*, *process*, *function*, called class words, represent generic hyperonyms characterizing definitions, in particular copula definitions (Pearson, 1998). Definitions that explain general concepts should present a high number of these generic words as hyperonyms, while definitions trying to explain domain specific concepts should make use of a high number of domain specific words as hyperonyms. For example, we can observe the two following definitions:

A fungus is a member of a large group of eukaryotic organisms that includes microorganisms such as yeasts and molds.
A mycotoxin is a metabolite produced by organisms of the fungi kingdom, commonly known as molds

The terms *fungus* and *mycotoxin* are both related to the domain of Botanics, but the term *fungus* represents a more generic concept in comparison with the term *mycotoxin*. When these two terms are defined, in the first case, a generic hyperonym with the function of a class word, such as *member of*, is used; in the second case, a very domain specific term, such as *metabolite*, is used.

Starting from this observation, we collected the first noun occurring after the connector verb *to be* in each copula definition. Table 6.3 shows the number of different nouns after the main verb, using the two algorithms and using as top category *Fundamentals* and *Main Topics*. The idea is that a smaller number of nouns corresponds to a more generic corpus, as the same class word is used several times in different domains. An increase in the number of different nouns is linked to an increase in the presence of

domain specific hyperonyms, and thus to a possibly less homogeneous corpus.

	EN							PT						
	Fundamentals			Topics			Mean	Fundamentals			Topics			Mean
	*Alg*1	*Alg*2	Mean	*Alg*1	Alg 2	Mean		*Alg*1	*Alg*2	Mean	*Alg*1	Alg 2	Mean	
1000	361	375	368	355	359	357	363	367	368	368	393	358	376	372
10000	1889	2057	1973	1941	1983	1962	1968	1630	1575	1603	1900	1713	1807	1705
25000	3274	3249	3262	3340	3210	3275	3268	2472	2493	2483	2472	2835	2654	2568
50000	4864	4845	4855	5127	5038	5083	4969	4039	3765	3902	4043	3474	3759	3830
100000	7811	7452	7632	8010	7734	7872	7752	5016	4926	4971	5658	5388	5523	5247

Table 6.3: Class word statistics for copula definitions

It is possible to note that even if the two Wikipedias differ greatly in terms of size, the number of terms characterizing definitions is practically the same, at least for corpora with 1,000 definitions. As the corpora get bigger the number of terms increases for each corpus, but for English this increase is larger. Nevertheless, the difference between English and Portuguese is quite insignificant with respect to the difference in terms of corpus size.

In general, *Alg*2 produces less terms, particularly for bigger corpora (25000 or above) when compared with *Alg*1. The same phenomenon can be observed with corpora built using *Fundamentals* instead of *Main Topics* as the top category. This result is more evident for Portuguese than for English. These observations suggest that *Alg*2 and *Fundamentals* produce more homogeneous corpora.

In order to confirm this first consideration, the terms extracted were ordered from the more to the less frequent. The idea is that in the top positions we expect to find class words such as those enumerated by Pearson (1998). If specific words appear, this means that the corpus over-represents a specific domain. For space reasons, we present only the first 25 terms for each algorithm and for each top category and for each corpus size (omitting the bigger corpus composed by 100,000 articles). Terms belonging to specific domains are underlined.

Tables 6.4, 6.5, 6.6, 6.7 show results for English. Regarding corpora with size 1000 and 10000, for both the algorithms and both top categories, the number of domain specific terms is very low (1 or 2). With bigger corpora the best results are obtained when *Fundamentas* is used instead of *Main Topics* and *Alg*2 instead of Alg1. Looking at the specific terms, we can see that when the *Fundamentals* category is used the

1,000	10,000	25,000	50,000
term	term	term	term
element	process	process	process
study	form	type	organization
name	element	name	plant
form	type	organization	name
concept	concept	form	type
process	name	plant	form
phenomenon	study	concept	concept
group	organization	element	method
type	method	method	study
state	theory	study	compound
model	system	system	element
theory	phenomenon	theory	species
statement	group	book	genus
organization	set	group	theory
method	model	genus	system
field	field	species	book
system	plant	set	group
act	approach	compound	set
ability	state	field	act
word	branch	branch	branch
principle	measure	phenomenon	research
part	part	practice	practice
meson	book	model	technique
genus	act	research	field

Table 6.4: English top words obtained with *Alg*1 algorithm and the category *Fundamentals*

1,000	10,000	25,000	50,000
term	term	term	term
concept	organization	organization	organization
form	process	process	process
study	form	form	type
process	type	type	form
organization	concept	name	name
theory	name	concept	concept
state	element	study	method
element	study	method	study
type	method	system	theory
phenomenon	theory	group	book
name	system	book	group
group	group	theory	system
approach	book	element	plant
act	field	plant	set
ability	act	set	act
system	state	field	field
science	set	act	practice
part	practice	research	branch
material	model	approach	research
book	research	practice	element
practice	branch	branch	business
model	phenomenon	state	approach
emotion	plant	phenomenon	technique
body	approach	movement	movement

Table 6.5: English top words obtained with *Alg*2 algorithm and the category *Fundamentals*

domains that are overrepresented are linked to the editorial area (book and journal) and to the botanical area (plant). When using *Main Topics*, at least other two over-represented domains are added, that is computer science (computer, software, language) and business (business and company).

Tables 6.8, 6.9, 6.10, 6.11 present the word lists for the Portuguese corpora. As for English, the best results are obtained when *Alg*2 is used in conjunction with *Fundamentals* category. Regarding over-represented domains, the situation is worse. As for English, we have the editorial area (revista ="magazine"), but then we have also the health field (*doença*="illness"), the astronomic domain (*asteróide*="asteroid", *galáxia*="galaxy"), the mathatics domain (*espiral*= "spiral", *número*="number"). When analyzing the word lists for *Main Topics*, we find again the computer science domain (*computador*="computer", *língua*="language"), but then we have also a number of other terms indicating very different domains such as *jogo*="game", *dia*="day", *freguesia*="municipality", etc.

These lists of words allow us to draw some final observations. In general corpora extracted starting from *Main Topics* are most affected by over-represented domains, especially when considering the three biggest corpora. This can be explained by the fact that this category has 22 children, representing specific domains. It is more likely

1,000	10,000	25,000	50,000
term	term	term	term
study	process	process	process
process	form	organization	organization
system	organization	form	form
set	study	type	type
research	type	name	name
branch	method	study	method
form	name	method	list
concept	concept	concept	study
computer	computer	list	book
practice	system	computer	concept
organization	field	book	journal
theory	branch	field	plant
period	research	system	computer
method	theory	practice	language
application	language	language	system
word	art	theory	research
type	technique	research	device
name	practice	art	practice
language	science	group	group
field	set	journal	theory
event	group	branch	art
discipline	act	set	field
act	book	technique	technique
state	device	area	set

Table 6.6: English top words obtained with *Alg*1 algorithm and the category *Main Topics*

1,000	10,000	25,000	50,000
term	term	term	term
study	process	process	organization
system	type	organization	process
process	form	type	type
method	method	form	form
practice	organization	method	method
organization	study	study	name
field	concept	concept	study
concept	name	name	concept
application	business	practice	plant
research	practice	research	book
form	system	field	device
branch	field	system	language
act	branch	business	journal
technology	research	set	system
technique	set	device	research
set	theory	theory	company
business	science	result	practice
theory	result	approach	act
ability	act	branch	list
type	device	technique	business
time	approach	language	set
science	technique	company	group
measure	language	group	technique
event	technology	act	software

Table 6.7: English top words obtained with *Alg*2 algorithm and the category *Main Topics*

1,000	10,000	25,000	50,000
termo	espiral	espiral	espiral
nome	galáxia	galáxia	asteroide
conjunto	termo	termo	galáxia
conceito	número	nome	espécie
forma	nome	espécie	nome
símbolo	espécie	tipo	termo
processo	tipo	organização	tipo
organização	doença	doença	organização
número	conjunto	conjunto	sistema
fenômeno	forma	forma	forma
sistema	organização	número	conjunto
tipo	processo	asteroide	empresa
teoria	sistema	processo	processo
expressão	conceito	sistema	doença
estado	ramo	grupo	número
designação	grupo	conceito	grupo
revista	asteroide	ramo	ramo
ramo	movimento	empresa	unidade
movimento	estrutura	expressão	órgão
unidade	área	gênero	instituição
palavra	designação	unidade	conceito
espécie	gênero	estrutura	movimento
parte	método	área	expressão
estudo	ciência	parte	instrumento
ato	estudo	método	programa

Table 6.8: Portuguese top words obtained with *Alg*1 algorithm and the category *Fundamentals*

1,000	10,000	25,000	50,000
termo	termo	espiral	espiral
forma	organização	galáxia	galáxia
conceito	nome	termo	asteroide
nome	número	nome	nome
conjunto	forma	organização	termo
processo	tipo	tipo	espécie
organização	conjunto	forma	empresa
movimento	espécie	conjunto	tipo
sistema	conceito	número	organização
tipo	processo	sistema	sistema
estado	sistema	espécie	forma
estudo	movimento	processo	conjunto
designação	ramo	doença	'orgão
área	doença	empresa	processo
parte	expressão	conceito	grupo
palavra	associação	ramo	instituição
fenômeno	grupo	grupo	doença
símbolo	instituição	movimento	unidade
revista	teoria	instituição	ramo
ramo	designação	'orgão	número
prática	empresa	expressão	movimento
método	área	associação	conceito
denominação	estudo	unidade	expressão
teoria	ato	asteroide	programa
'orgão	ciência	área	associação

Table 6.9: Portuguese top words obtained with *Alg*2 algorithm and the category *Fundamentals*

1,000	10,000	25,000	50,000
termo	termo	nome	nome
nome	nome	termo	termo
conjunto	tipo	tipo	espécie
sistema	conjunto	sistema	tipo
forma	sistema	conjunto	género
conceito	forma	forma	sistema
tipo	processo	espécie	espiral
processo	ramo	processo	conjunto
computador	computador	jogo	forma
área	língua	doença	jogo
ramo	organização	espiral	organização
ciência	conceito	organização	instituição
técnica	espécie	ramo	processo
programa	método	língua	empresa
organização	expressão	conceito	língua
palavra	dispositivo	empresa	programa
expressão	designação	programa	galáxia
método	movimento	expressão	grupo
estudo	estudo	método	número
documento	ciência	dispositivo	ramo
dispositivo	área	movimento	doença
designação	programa	instituição	conceito
tecnologia	técnica	designação	escola
revista	empresa	computador	método
instituição	instituição	grupo	instrumento

Table 6.10: Portuguese top words obtained with *Alg*1 algorithm and the category *Main Topics*

1,000	10,000	25,000	50,000
termo	termo	nome	género
conjunto	nome	termo	nome
forma	conjunto	tipo	termo
processo	sistema	sistema	tipo
nome	tipo	conjunto	sistema
sistema	forma	forma	empresa
tipo	processo	processo	espécie
designação	ramo	organização	organização
computador	conceito	ramo	conjunto
técnica	organização	empresa	espiral
revista	computador	conceito	forma
organização	língua	programa	processo
expressão	dispositivo	espécie	grupo
conceito	movimento	doença	gênero
área	método	dispositivo	ramo
ramo	empresa	número	número
grupo	expressão	movimento	jogo
ato	estudo	método	galáxia
dispositivo	designação	dia	dia
ciência	área	designação	instituição
tecnologia	técnica	instrumento	doença
programa	doença	língua	programa
estudo	ciência	grupo	movimento
estrutura	parte	expressão	conceito
palavra	programa	computador	método

Table 6.11: Portuguese top words obtained with *Alg*2 algorithm and the category *Main Topics*

to encounter an over-specified area, composed for example by a list of all galaxies or of all plants.

When comparing the English and Portuguese languages, Portuguese corpora present a larger number of over-represented domains. A possible explanation for this takes in consideration the size of Wikipedia, as Portuguese Wikipedia is by far smaller than the English one, the number of articles on general topics runs out sooner.

6.4 Machine Learning Experiments

With the knowledge gained in the previous Section, we can now extract the definitions we need to build a new dataset to classify verbal definitions, using *Alg*2 applied to *Fundamentas*.

The new dataset will consist of all the positive and negative examples of the dataset used in the previous Chapter, plus a number of new positive examples to end up with the same number of positive and negative examples, resulting in a fully balanced dataset. In the case of the Portuguese language, the original dataset was composed by 744 negative examples and 98 positives ones. In order to obtain a balanced dataset, 646 new positive examples were added. The original English dataset was composed by 3516 negative

examples and 124 positive examples. For this, 3392 new positive examples were added.

Similarly to the experiments with verbal definitions presented in the previous Chapter, we used as feature part-of-speech 3-grams after the connector, as we verified that this setting ensures the best results for verbal definitions. In previous experiments with verbal definitions, we added to the feature list all the connector verbs present in the corpus. In the new experiments with Wikipedia, we also tested how to exploit Wikipedia to improve the feature selection of the new classifiers. In particular, the 25 most frequent verbs with prepositions were collected for each language, creating a new list of features.

Table 6.12 shows the list of verbal patterns extracted from Wikipedia for both languages.

ENGLISH	PORTUGUESE
to refer to	referir -se
to know as	afirmar que
to be defined as	caracterizar -se
to be used to	chamar -se
to consist of	consistir de
to occur when	consistir em
to be described as	corresponder a
to be composed of	denominar -se
to be used in	designar -se
to name after	dizer -se
to be found in	entender -se
to be established in	estabelecer que
to be related to	ocorrer quando
to be know in	referir a
to be used by	ser caracterizado por
to occur in	ser composto por
to stand for	ser definido como
to be based on	ser formado por
to involve in	ser fundado em
to be used for	ser utilizado em
to be used as	ser utilizado para
to be concerned to	ter como
to describe how	tratar -se
to be characterized by	vir de
to be divided into	ser conhecido por

Table 6.12: List of verbal expressions gathered from Wikipedia

Three different settings were tested. One where only the first list of verbs was used. The second were only the list gathered from Wikipedia was used. And a final one, where the two lists were combined.

The three classifiers with better results are presented for each experimental setting

(Table 6.13). To compare classifiers, the F-measure metric is used. In general, classifiers present a better performance when both connectors verbs and verbal patterns from Wikipedia are used in combination. In particular, when only Wikipedia expressions are used, the performance is worst than when only connector verbs are used. When the two lists are used in combination, the best results are obtained. Only in the case of the VFI algorithm applied to English, this is partially true, as in this case the Wikipedia list gets a better F-measure score than the connector verb list, but again, the combination of both obtains the best performance.

	Portugese				English			
	P	R	F-m	AUC	P	R	F-m	AUC
				Naïve Bayes				
Verbs	0.81	0.84	0.83	0.90	0.92	0.92	0.92	0.96
Wikipedia	0.80	0.82	0.81	0.90	0.92	0.90	0.91	0.96
Both	0.85	0.83	0.84	0.91	0.93	0.91	0.92	0.96
				Random Forest				
Verbs	0.76	0.75	0.75	0.84	0.92	0.91	0.91	0.96
Wikipedia	0.79	0.71	0.75	0.83	0.90	0.90	0.90	0.95
Both	0.81	0.75	0.78	0.86	0.92	0.91	0.91	0.97
				VFI				
Verbs	0.62	0.94	0.75	0.85	0.65	0.96	0.78	0.92
Wikipedia	0.89	0.49	0.63	0.82	0.88	0.76	0.82	0.92
Both	0.84	0.69	0.76	0.86	0.81	0.88	0.85	0.92

Table 6.13: Results for verbal definitions using Wikipedia

For Portuguese, the best algorithm is Naïve Bayes followed, at a certain distance, by Random Forest and VFI. These two last algorithms present very similar results. For English, this classification is confirmed, but in this case, Naïve Bayes and Random Forest present very similar results, while VFI remains far behind.

In general, the performance of all the classifiers is better than the one obtained with the sampling of the dataset showed in the previous Chapter, but for English this improvement is much more evident (see below). When sampling techniques were used, the best result for Portuguese was an F-measure of 0.77, obtained with a combination of Tomek Links and SMOTE with Naïve Bayes, while for English the best results was an F-measure of 0.78, with the same combination of algorithms. With the use

	P	R	F-m	AUC
Random Forest	0.84	0.76	0.81	0.89
Naïve Bayes	0.92	0.72	0.80	0.97
VFI	0.91	0.15	0.26	0.85

Table 6.14: Results for English classifier with a reduced dataset

of Wikipedia, the best result for Portuguese reaches an F-measure of 0.84, while for English, an F-measure of 0.92.

The difference in performance between the two languages is probably due to the dataset size, in particular the number of positive examples, which were 3516 for English and 744 for Portuguese. This may suggest that the Portuguese dataset does not have a sufficient number of examples to cover the multiplicity of different structures characterizing verbal definitions and, consequently, to train a good classifier. As it was quite hard to increase the number of Portuguese data set, as it would need inclusion of negative examples, gathered by human annotators, this hypothesis was tested with the English dataset, reducing the number of negative examples to 750 and consequently the number of negative examples.

Table 6.14 presents results for the three classifiers when the reduced dataset was used. These results are better than the ones obtained with sampling techniques without Wikipedia, but, as we expected, they are quite worst than the ones obtained with the entire dataset.

We also tested our strategy by means of a different evaluation method beside the 10-fold cross validation. As we did with copula definitions, we split each corpus in two sub-corpora, a development corpus, consisting of 75% of the whole corpus used for training the classifier and an held out 25% of the corpus, used for testing the classifier. Regarding the Portuguese corpus, the division in two sub-corpora follows the division carried out in the development of the rule-based system.

In Table 6.15, we present the best results obtained with the three different approaches, that is rule-based, machine learning with 10-fold cross validation and machine learning with a separate testing corpus.

Similarly to the results obtained with copula definitions, we observe a slight worsening in the results obtained with the testing corpus in comparison with the 10-fold cross validation strategy. This time, the deterioration affected only the Portuguese corpus.

	Rule Based			10-fold			Testing Corpus		
	P	R	F-m	P	R	F-m	P	R	F-m
PT	0.14	0.65	0.23	0.85	0.83	0.84	0.76	0.71	0.73
EN	-	-	-	0.93	0.91	0.92	0.93	0.88	0.91

Table 6.15: Results using different strategies with verbal definitions

Again, in order to test if this was due to the fact that the 10-fold cross validation uses 90% of the dataset for training the classifier, while our testing corpus only use the 75% of the corpus, we increased the testing corpus to 90%. In this case, the results for the Portuguese language were very similar to the ones obtained with the 10-fold cross validation.

6.5 Conclusions

In this Chapter, a method to improve the performance for verbal definition extraction, using Wikipedia, was presented. Exploiting the structure of this on-line encyclopedia, we described a method for building a corpus of definitions using Wikipedia, applicable to different languages.

As Wikipedia is developed without a predetermined design, it happens that some knowledge fields are overrepresented, containing a high number of very specific articles, which can lead to a corpus that is biased to some domain and can not be considered a general corpus. The first issue to deal with regards how to select articles to be used for the construction of a corpus containing general definitions.

We have tested two different algorithms and two different starting point categories from which it is possible to extract articles over corpora of different sizes. The use of the entire Wikipedia is not recommended for two main reasons. Firstly, its size would make the processing very hard and slow and, furthermore, for this kind of task, we do not need a huge number of articles. Second, if we use the entire Wikipedia, it is impossible to ensure a balanced corpus, as some knowledge fields in Wikipedia could be over represented.

The algorithm that favors the extraction of articles on the same level of depth in the category tree structure, in combination with the *Fundamentals* category, obtained the best results in terms of corpus balance.

This algorithm was used to extract definitions in English and Portuguese in order to balance the original respective corpora. Wikipedia was also used to extract verbal patterns to be used as features in the construction of classifiers. The best performance was obtained, for both languages, with Naïve Bayes classifiers and with the inclusion in the features list of the verbal patterns collected from Wikipedia.

Results for both languages outperform results obtained with the application of sampling techniques and they also outperform the state of the art results.

Chapter 7

Glossary Construction for e-Learning

7.1 Introduction

During the development of this doctoral research work, the opportunity to test the definition extractor tool in a concrete situation came up. Thanks to this, a qualitative evaluation together with quantitative evaluation was carried out. The definition extractor was integrated in a Learning Management Systems (LMS) in order to support the task of glossary creation in an e-learning environment.[1]

An LMS is a software system that provides the necessary tools for managing and creating courses, and for scheduling training or e-learning activities in an organization. It manages the delivery of self-paced, e-learning courses, and allows the publication of courses, putting them in an on-line catalog. Learners log into the LMS, select courses from the catalog and launch them. Typically, an LMS provides numerous functionalities such as a submission management system, file storage, bulk mail, groups management, fora, calendars, news, surveys, assessments, etc. In the last years, we have witnessed the increasing adoption of Learning Management Systems (LMS) by different organizations, from universities to companies, while new LMS's keep coming up in order to meet increasing needs.

One of the main issues in the use of an LMS is the creation of rich metadata information for each piece of learning content, also known as Learning Objects (LO), in order to allow an effective reuse and support the learning process. Basically, LOs are

[1]This Chapter is based on Del Gaudio & Branco (2009a) and Del Gaudio & Branco (2009b)

units of study, exercise, or practice that can be consumed in a single session, and they represent reusable modules that can be authored independently of the delivery medium and be accessed dynamically.

Ideally, LOs can be exchanged between different LMSs and plugged together to build classes that are intended to serve a particular purpose or goal. Learning objects are typically described as a collection of metadata that facilitates reusability. Without good metadata and an effective retrieval tool, the reusability of an LO is limited because it will be difficult or impossible to identify and retrieve it from the bulk repositories of LOs. Furthermore, LOs can be improved in order to support the learning process, by inserting new information such as glossaries. Despite the fact that glossaries have been found useful in the learning process, facilitating reception of texts and acquisition of knowledge (Weiten *et al.*, 1999), many LOs are not enriched with this resource because the process of producing glossaries is very time-consuming.

This work was developed within the european funded project Language Technology for e-Learning (LT4eL)(Monachesi *et al.*, 2006). The aim of this project was to develop tools using language technology techniques to support e-Learning applications in a multilingual environment. The languages represented in the project were Bulgarian, Czech, Dutch, German, English, Polish, Portuguese, and Romanian.

In the project LT4eL different tools were developed in order to support two different tasks: metadata creation and retrieval of Learning Objects. Specifically, Language Technology resources and tools were employed for the semi-automatic generation of descriptive metadata. New functionalities were developed such as a key-word extractor and a glossary candidate detector, tuned for the various languages addressed in the project (Monachesi & Westerhout, 2008).

For evaluation and validation purposes, the LT4eL tools and resources have been integrated into the LMS ILIAS. In particular, the ILIAS Learning Management Systems was extended with new functionalities to support the different actors in e-learning environments. ILIAS is a full fledged web-based learning management system that allows users to create, edit and publish learning and teaching material in an integrated system with their normal web browsers[1].

The main objective of this evaluation was to provide evidence regarding the measure in which an automatic definition extractor can enhance the e-learning process, for both

[1] www.ilias.de

students and content providers or tutors. The evaluation of the improved LMS is focused mainly on the effects on the students' learning process and secondarily on the tutors experience in generating metadata, in particular glossaries. Through qualitative and quantitative evaluation we demonstrate the improvement of such new functionalities in the learning experience.

The evaluation carried out within the Lt4eL project covered all the new functionalities developed within the project. Here we report the results regarding the specific impact of a definition extractor on the performance of students and tutors.

7.2 The Learning Management System ILIAS plus Definition Facilities

The LMS chosen for improvement, ILIAS, is a powerful open source web-based learning management system that enables the to easy management of learning resources in an integrated system via a Web Service Interface (SOAP). Some of the features supported are course and group management, Individual Personal Desktop, Repository with Role Based Access Control, Learning Progress Management, Test and Assessment, Chat, Forums, Exercises. Furthermore, it complies with LOM (Learning Object Metadata) and SCORM (Shareable Content Object Reference Model), standards for e-learning content.

In order to support glossary building, the definition extractor derived from the rule-based module was integrated in ILIAS by the project LT4eL. Using this tool, tutors can select an LO and automatically generate a list of possible definitions, that can be included in the glossary for that particular LO. Definitions extracted in this way can be filtered out and extended by the user. Again, this functionality can be used by tutors in their task of meta-data creation to speed up the process. Students can use it as well, in order to obtain a draft overview of the concepts being defined in an LO imported into the LMS. Figure 7.1 shows the interface presented to tutors when they use the automatic glossary construction function.

The term defined appears in a separate box, then the candidate definition sentence is presented and finally the context where the sentence is displayed. It is possible to edit the information contained in the box with the definition sentences, in order to reformulate the definition.

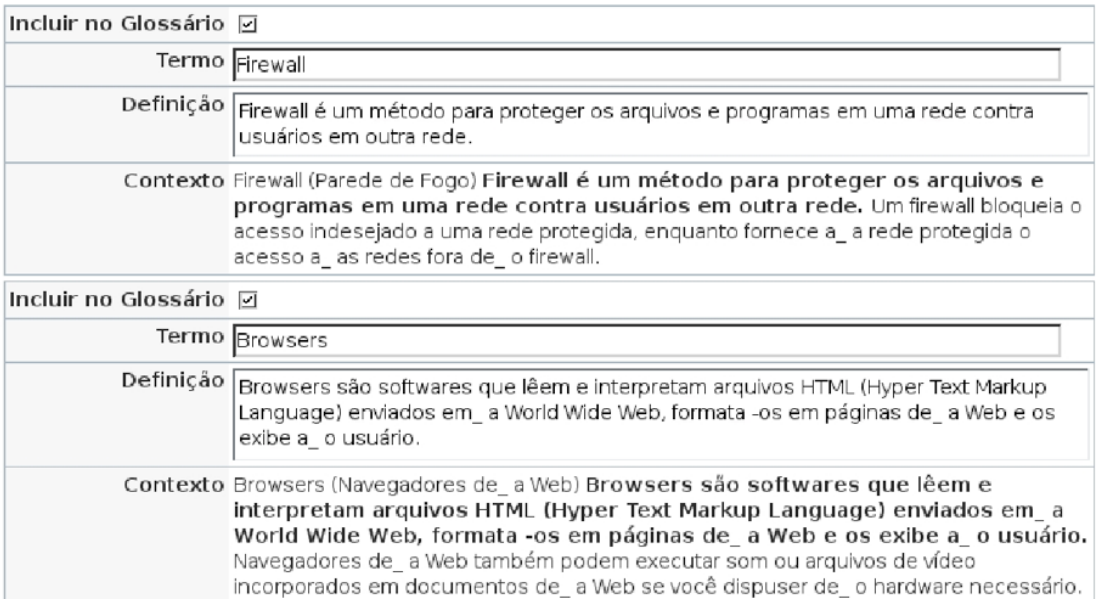

Figure 7.1: The print-screen of the interface of the definition extractor implemented in ILIAS

Furthermore, the ILIAS search engine was improved in order to allow searches for definitions. With this new search facility it was possible to evaluate the value of the availability of definitions in the learning process for students. Figure 7.2 shows the interface presented to students when they search for a particular concept. They can select different languages and different search methods, besides definitions. In particular, three other different search methods are available: text, keyword and semantic and conceptual.

The first one is the text search, that is already offered in the ILIAS common distribution, and consists in a search for a specific word present in the document. The keyword search is based on a tool that automatically generates keywords for a LO and then allows searching for a specific keyword. Finally, a Semantic and Multilingual Search Tool was developed. A key component of this tool is an ontology and the annotation of the Learning Objects with their concepts. In the LOs, each natural language expression conveying one of those concepts is associated to such concept via metadata annotation. Accordingly, the search tool permits to retrieve Learning Objects given the concept entered and its occurrence in the retrieved objects. Since the ontology is common for LOs from different idioms, the set of retrieved objects can include also those not written in

Figure 7.2: The print-screen of the interface of research facility, with the possibility to look for definitions of a given concept

the user's language, thus cross-language search is supported. Based on this ontology, two different search facilities were integrated in the LMS, the semantic search and the conceptual search. In the semantic search, the user inserts a world in the search box, and the system searches for the word in the ontology and retrieves the LOs based on the corresponding concepts. In the case of conceptual search, the user is presented with a list of similar concepts, hyperonyms and hyponyms, and he selects which concepts to include in the search.

7.3 Scenario Based Evaluation

The main objective underlying this experiment was to show how the automatic definition extractor, integrated in an LMS, enhances the e-learning experience, increasing the effectiveness of learning and teaching. This evaluation was designed in order to get some insight on the satisfaction of the potential end-users with respect to the new functionalities and it was based on the user scenario methodology (Carrol, 1995). "Scenario" here is meant to be "a story focused on a user, which provides information on the nature of the user, the goals he wishes to achieve and the context in which the activities will take place".

We developed scenarios for two different roles (tutors and students) in order to

evaluate the impact of the new tool along the whole learning process. In both scenarios the improved ILIAS with definition extraction (ILIAS-DE) was compared with ILIAS standard version (ILIAS-ST), that is ILIAS without the new functionalities.

7.3.1 Tutor Scenario

For this evaluation scenario, we selected 10 professors of the Department of Informatics of the University of Lisbon. Participants assumed the role of a tutor who would be teaching a short course covering “Information Exchange over the Internet" to first year students in humanities at their university.

Tutors were divided in two groups, both groups using the same technologies, that is ILIAS-ST, ILIAS-DE and internet, but in different orders. Tutors were provided with familiarization sessions about ILIAS functionalities.

Participants were presented with an LO in the LMS and were requested to generate a glossary using the tools in order to make the LO available for a particular course. They were asked to generate the glossary manually, using ILIAS-ST, or they were allowed to use the internet, or they had access to ILIAS-DE. The time they took to complete the tasks in the tree different situations was recorded. When using ILIAS-DE, they needed on average around half the time to generate the glossary than when they had no kind of help. Also, they experienced no big improvement with the use of internet.

Furthermore, tutors were asked for a qualitative appreciation of the improved system. They were asked to give a qualitative evaluation of the tool used. All testers (100% of score) agreed that the definition extractor was very useful to complete their task, and said they would use it if it was available in a regular distribution of the LMS.

7.3.2 Student Scenario

In the student scenario, students used multiple-choice questions as the main instrument to interact with the system, i.e. they were presented with a question and possible answers, and then encouraged to use specified resources to find the right answer. Multiple-choice assessments are often used to determine the level of comprehension of written texts and may also be used to assess levels of knowledge and understanding at a specific moment in time. It is an assessment method that can be administered and graded electronically, lending to situations where candidates are spread geographically and tests are not held simultaneously. Such types of tests are the most frequent way

of assessment in e-Learning systems, as they allow automatic greding. Multiple-choice assessments would therefore support our requirement for quantitative analysis and for comparison of scores.

Students were asked to take on the role of a first year undergraduate student, studying a short course covering “Information Exchange over the Internet". They were told that their tutor had prepared some research questions for them in the form of a quiz to enable them to gain a basic grounding in key aspects. The students were divided into 4 groups. Groups 1 and 2 formed the target groups and Groups 3 and 4 formed the control groups. Each group was asked to answer two sets of questions, set A and set B. Group 1 answered set A using ILIAS-ST and set B using ILIAS-DE. For group 2 the order of questions in the sets were inverted, so that, the set B of questions was answered with ILIAS-ST whereas the set A questions where answered with ILIAS-DE. The same sets of questions were used for the control groups, but in this case both groups answered the questions without resorting to LOs in ILIAS, but to information in web facilities such as Google, Wikipedia, etc.

In this way, two elements of control were introduced in order to objectively test the hypotheses relating to new functionalities. Two control groups were set up which would use only internet searches to help them answer the quiz questions. For the second control method, the questions were split into two sets and each set answered by half the students using ILIAS-ST functionality and the remainder by those using ILIAS-DE.

To provide the basis for carrying on the scenario-based evaluation, we conducted Pre- and Post-tests to assess the level of knowledge both before and after the quiz being solved by the students. These tests consisted of two sets of conceptual mapping and terminology questions to answer without reference to content. The answers provided by the students to all the questions were collected electronically.

The questions within each set were of similar structure and designed to encourage students to research both terminology and relationships between concepts as they searched for information to help them select an answer. Each question within the experiment was timed by the system. This arrangement offered us with a number of potential statistical comparisons to test hypotheses.

The participants in this evaluation were 24 undergraduate students, arranged in the following way: 16 students in the target groups, respectively 8 in group 1 and 8 in group 2, and 8 student in the control groups, respectively 4 in group 3 and four in group 4.

Group 1		Group 2	
Question	Technology	Question	Technology
Pre-test	-	Pre-test	-
Set A	ILIAS-ST	Set B	ILIAS-DE
Set B	ILIAS-DE	Set A	ILIAS-ST
Post-test	-	Post-test	-

Group 3		Group 4	
Question	Technology	Question	Technology
Pre-test	-	Pre-test	-
Set A	Internet	Set B	Internet
Set B	Internet	Set A	Internet
Post-test	-	Post-test	-

Table 7.1: Student scenario structure

Before each section quiz, a familiarization session was carried out. Before starting the experiment, the general structure of the experiment was explained, and then students were asked to answer the pre-test questions. For group 1, the second familiarization was about ILIAS general functionalities, and after that, students were asked to answer a quiz using ILIAS. The third session was about the new functionalities introduced in the LMS, and then the students were asked to answer the quiz using ILIAS-DE. Table 7.1 shows the experimental settings for all four groups.

7.3.2.1 Student Scenario Results

In a first data analysis carried out on the data gathered form the student scenario, we found that students in the target group using the ILIAS-DE (group 1 and 2) gave the correct answer more frequently than the students in the other two groups. As showed in Table 7.2, testers using the ILIAS-DE gave, on average, the correct answer to more than 5 questions over a questionnaire composed by 7 questions. The control group obtained the second best result, with 4.9, followed by the group using ILIAS standard version.

Comparing the number of correct answers in pre-test and post-test of the different groups, we noticed that the largest improvement was obtained by the target group. In Table 7.3 the average number of correct answers over a questionnaire composed by 5 questions is showed.

Using the new functionality, learners are supported towards a better, more effective grasp of the terminological and conceptual space that defines a certain domain of knowledge.

Test	Group	Score
Pre-Test	Control	3.28
Post-Test	Control	4.66
Improvement	Control	1.38
Pre-Test	Target	3.11
Post-Test	Target	4.83
Improvement	Target	1.72

Table 7.2: Mean scores for control end target groups

Group	Technology	Score
Target Group	ILIAS-DE	5.27
Target Group	ILIAS	4.72
Control Group	Web Search	4.90

Table 7.3: Pre-test and post-test scores for control end target groups

Group	Technology	Mean Time *(Sec.)*
Target Group	ILIAS-DE	34.97
Target Group	ILIAS	82.23
Control Group	Web Search	58.60

Table 7.4: Mean time in seconds for answering a question for control end target groups

Search method	%
Full Text	21%
Keywords	29%
Semantic	7%
Concept	7%
Definition	36%

Table 7.5: Student satisfaction for each search method

Comparing the time taken on average to correctly answer the quiz, the best result was obtained by the students using the ILIAS-DE. Table 7.4 presents the average time in seconds used to answer a question for each group and for each technology. This means that the tools we developed are able not only to improve the learning process in terms of knowledge acquisition but also in terms of the time spent in this process.

Besides quantitative results, qualitative results were also gathered. In particular, we asked students using ILIAS-DE to compare the different search methods implemented in the system, indicating which search method they found most useful. They were asked to pick the single functionality they considered most useful. Table 7.5 shows the user satisfaction for each search facility. Definition search was the method more appreciated by students, followed by keyword search. Semantic and concept search do not satisfy students.

For each search method, we asked students for their opinion regarding the usefulness of that specific search facility for their study. In particular, after completing the Post-test, for each of the search method in turn, they were asked to indicate their degree of agreement with the statement: "the *target* search would be useful for my studies". Table 7.6 shows the results of this test. As for the previous judgements, students, in general, appreciate the definition search, even if, in this case, they judged the keyword

Search Method	Agreed	Disagreed	Neutral
Full Text	60%	7%	33%
Keywords	86%	7%	7%
Semantic	60%	13%	27%
Definition	80%	7%	13%
Concept	26%	20%	54%

Table 7.6: Students judgment of usefulness for each search method

search a little bit more useful.

7.4 Conclusion

This evaluation experiment aimed to demonstrate the importance of the definition extractor in a real application and at the same time to extrinsically evaluate the quality of the developed extractor. The user scenario evaluation allowed the test of the definition extractor in a real application. Users are reacting positively to the functionalities that have been added to ILIAS and that we are probed with the scenarios.

Using this new added functionality, a content provider can select a document and automatically generate a list of possible definitions that can be included in the glossary for that particular document. Definitions extracted in this way can be filtered out and extended by the users. This functionality is used by tutors in their task of glossary generation and helps to speed up that process. Results presented here show that the definition extractor halves the time needed to create a glossary for a given document when an LMS is used.

Regarding student experience, results support the initial hypothesis that learners using the new functionality are more effective in grasping the terminological and conceptual space around the learning topics, as they can use the definition extractor in order to obtain a draft overview of the concepts being defined in an LO imported into the LMS.

Chapter 8

Conclusions

8.1 Introduction

The present work addressed the problem of automatic definition extraction from unstructured texts. Two main objectives were pursued and achieved. The first objective was to provide a set of methods and heuristics for building definition extractors that can be applied to different languages of the corpora or domain at stake. The second objective was to build a generic definition extractor for the Portuguese language.

This research focused on three types of definitions, characterized by the connector between the defined term and its description, namely copula, verbal and punctuation definitions. The strategy adopted can be seen as a "divide and conquer" approach. Differently from other works so far, specific heuristics were developed in order to deal with different definition typologies, whose effectiveness is independent. We used rule-based methods to extract punctuation definitions, machine learning with sampling algorithms for copula definitions, and machine learning with a method to increase the number of positive examples for verbal definitions.

For each type of definition, the portability of the proposed definition extractor is ensured by means of selecting rules, features, and other external resources that are as much as possible domain and language independent. The methods and heuristics developed can easily be ported to other languages to obtain new definitions extractors in a few automatic or semiautomatic steps. These design features represent a major advance in the field of definitions extraction, as other previous works in the literature have focused on specific corpora or specific domains, whose results are hardly reproducible when the domain or the language change.

This final Chapter reviews the work carried out and considers the results achieved. For each definition type, we look at experiments separately, and address what was achieved.

Finally, we conclude by giving a direction for further experiments and work for definition extraction in general, based on the techniques used in this thesis.

8.2 Related Work

At the beginning of this dissertation, we discussed what a definition is, starting with a brief overview of concepts and theories that underpin the task of automatic definitions extraction, such as the distinction between word and term, on the one hand, and between general language and sublanguage and specialized language, on the other hand.

Specialized knowledge is associated with the activities and specific communicative situations of a specific area, in contrast with the general knowledge that is present in a wider range of situations shared by members of a community. The language used in these communicative situations is a specialized language or sublanguage, which can be analyzed more effectively than general language, as it is more restricted. The act of defining a term can be considered a sublanguage itself, subject to be analyzed, supporting, this way, the automatic extraction of definitions. Furthermore, the textual sources of definitions are documents in a specific knowledge area, and for this reason, are characterized by a specialized language, which is in turn more restricted compared to the general language.

The notion of definition was discussed, reporting the point of view of various scholars, mostly in terms of theoretical approaches. In particular, we presented the distinction between the so-called real and nominal definitions. The first one was addressed by the first philosophical reflections on the notion of definition, and states that a definition fully explains the nature of a concept. A nominal definition, on the other hand, explains the meaning of a word or a term.

Several authors have proposed other classifications based on the purposes of a definition and the methods by which these are expressed, with the two main types of definitions being represented by lexical and stipulative definitions. A lexical definition is the one found in a general language dictionary, reporting the meaning of a term as this is already used within a language community. A stipulative definition, on the other

hand, is a proposal or determination that a certain expression be used as a sign of a certain concept.

Regarding the method a definition is given, the most common is, by far, the analytical definition, by *genus* and *differentia*. This kind of definitions resemble an equation, following the schema $X = Y + C$, where X is the *definiendum* (what is to be defined), " $=$ " is the equivalence relation expressed by some connector, and the expression $Y + C$ is the *definiens* (the part which is doing the defining). The *definiens* should consist of two parts: Y is the *genus* (the nearest superior concept), the class of which X is an instance or a subclass, and C represents the *differentiae specificae* (the distinguishing characteristics) that turn X distinguishable from other instances or subclasses of Y.

Despite the possibility for a rich and fine-grained classification and analysis, when definitions are analyzed in their occurrence in texts, the focus tends to be on analytical definition and its possible variations. For example, we can have an exclusive *genus* definition, where only the superordinate concept is identified, and no description of the distinctive characteristics is provided. We can have definitions were the *genus* is omitted and the distinctive characteristics indicate the function or enumerate the parts of the concept defined. In this way, definitions may convey a specific semantic relation, such as hyperonimical, meronymic, causal and purpose.

When turning to the automatic extraction of definitions, most works are focused on the extraction of a definition in a sentence composed by a subject, a verb such as *to be*, *to mean*, *to define*, etc., and a predicative phrase. Some works take into consideration also verbal patterns indicating different semantic relations, such as causal, functional, etc. Few studies use other lexical patterns besides verbal ones, such as *is the term for*, *type of*, *kind of*. In general, the vast majority of the studies on definition extraction are based on a set of hand-crafted rules or patterns in order to identify definitions in texts. Some of the more recent works seek to improve the outcome of these rules by using machine learning techniques.

Since the 90s, there has been intense research activity around the extraction of definitional information. For instance, Hearst (1992) proposed a method to identify a set of lexico-syntactic patterns to extract hyponym relations from large corpora and extend WordNet with them. This method was adopted by Pearson (1996) to cover other types of relations.

One of the most effective systems, DEFINDER (Klavans & Muresan, 2001), combines simple cue-phrases and structural indicators introducing the definitions and the defined term. The corpus used to support the development of the rules consists of well-structured medical documents, where 60% of the definitions are introduced by a set of limited text markers. The nature of the corpus used can explain the high performance obtained by this system (0.87 precision and 0.75 recall).

Malaise *et al.* (2004) focused their work on the extraction of definitory expressions containing hyperonym and synonym relations from French corpora. These authors used lexical-syntactic markers and patterns to detect these two types of definitions. In this way, for hyponym and synonym definitions, they obtained, respectively, 0.04 and 0.36 of recall, and 0.61 and 0.66 of precision.

In Alarcón *et al.* (2009) a method for extracting definitions for the Spanish language called ECODE is described. It uses a broad corpus composed of over 1,000 documents covering eight different domains, namely Law, Human Genome, Economy, Environment, Medicine, Informatics and General Language. Basically, the system is composed by three modules. The first module automatically extracts the sentences by resorting to a pattern module composed by 15 definitional patterns constructed manually. The second module filters the output of the first one by applying a rule-based system. Finally, there is a third module that marks the *definiens* and the *definiendum*. The performance of the system was calculated for each different pattern, with an F-measure ranging form 0.45 to 0.95, and a mean of 0.72.

When machine learning techniques have been applied, the output of the pattern matching module has been used as the training dataset. When the pattern based module is characterized by a good performance in terms of recall but a poor performance in terms of precision, the machine learning module is used as a filter to discard false positive examples returned by the previous pattern matching step.

For instance, Westerhout & Monachesi (2008) combine syntactic patterns with a Naïve Bayes classification algorithm with the aim of extracting glossaries from tutorial documents in Dutch. They use several properties and several combinations of them, obtaining a precision of 0.80 and a recall of 0.78. This represents an improvement of precision of 0.52 points but a decline in the recall of 0.19 points in comparison with the syntactic pattern system developed previously by the authors using the same corpus.

Miliaraki & Androutsopoulos (2004) used a machine learning-based method to identify 250-character single-snippet answers to definition questions by using a collection of documents. They experimented with three different algorithms, namely Naïve Bayes, Decision Tree and Support Vector Machine (SVM), obtaining the best score with SVM with an F-measure of 0.83.

Fahmi & Bouma (2006) used a maximum entropy classifier. The corpus used was composed by medical pages of Dutch Wikipedia, from where they extracted sentences based on syntactic features. The dataset was composed by 2,299 sentences of which 1,366 were actual definitions. The initial accuracy of 0.59, obtained with the pattern based module, was improved with machine learning algorithms until it reached 0.92.

In very few cases, the machine learning algorithms were applied alone, skipping the pattern based step and without facing the problem of data imbalance. Chang & Zheng (2007) report on a system to extract definitions from off-line documents. As their corpus was composed by text snippets collected over the web, they end up with a quite balanced dataset.

The problem with the approach based on pattern matching is that it relies strongly on the set of manual crafted rules developed to ensure the first step of the process. Excluding the case of a few very general heuristics, whenever one needs to build a new system to extract definitions, it is necessary to start almost from scratch, by starting to analyze a possible set of definitions and then building a set of specific patterns. Furthermore, these rules are not only pertinent to a specific natural language, but also to a specific domain and application, making it difficult to extend their use beyond the constrained applicational context within which they were developed.

8.3 The Divide and Conquer Strategy

The strategy adopted in this work can be seen as a "divide and conquer" approach. By analyzing the work done in the field of automatic extraction of definitions, we have identified three types of definitions with increasing complexity. The classification is based on the type of connector that links the defined term and its explanation. The simplest type is the one where the connector is composed of a punctuation mark, such as the colon and the stretch, as in the example *TCP/IP: protocols used in the transfer*

of information between computers. For this kind of definition, a rule-based module was developed.

The second type of definition is characterized by the presence of the verb *to be* as a connector, such as in the sentence *FTP is a protocol that allows the transfer of archives from a place to another through the Internet.* This kind of definition was dealt with a module based on machine learning and sampling algorithms in order to balance the dataset.

The third type of definition present any verb except the verb *to be* as connector, as in the example, *An ontology can be described as a formal definition of objects.* In this case, the dataset has been enriched using an external source of definitions automatically extracted from Wikipedia.

While this classification is present in other works (see for example Westerhout & Monachesi (2007)), the novelty of our research relies on he fact that, for every type of definition, different techniques and specific heuristics have been applied.

8.3.1 Corpora

In this work, different corpora were used, in different languages, namely Portuguese, English and Dutch (this last one was only used for testing the machine learning experiments for copula definitions). All three corpora were collected in the context of the European project Language Technology for e-Learning LT4eL.[1]

These corpora cover the domains of Computer Science and eLearning and are encoded in an XML-based format which includes the linguistic annotation with part-of-speech (POS), lemma and morphological analysis information (automatically assigned).

Though covering different languages, these corpora are comparable as they were collected for the same purpose and using the same guidelines: they include learning materials written by experts for laypersons or relative experts on information technology. Furthermore, they are easily usable and the results of different experiments are easily comparable, given that they are annotated with the same type of morphosyntactic information across the different languages, and this information is encoded in a common XML format in all of them.

[1] www.lt4el.eu.

The sentences conveying definitions were manually annotated. In each such sentence, the term defined, the definition and the connection verb were annotated using a different XML tag.

The Portuguese corpus contains 23 tutorials and scientific papers in the field of Information Technology and has a size of 223,049 tokens and 10,941 sentences. It was automatically annotated with the LX-Suite (Branco & Silva, 2006).

The English corpus is a collection of 7 tutorials with a total size of 287,910 tokens and 20,172 sentences. The corpus was annotated with linguistic information, using the Stanford POS tagger (Toutanova & Manning, 2000).

The Dutch corpus is composed of 26 tutorials with a total size of 353,174 tokens and 23,996 sentences. The corpus was annotated with morphosyntactic features with the Wotan tagger and with lemmas provided by the CGN lemmatizer (Westerhout & Monachesi, 2007).

8.3.2 Punctuation Definitions

Punctuation definitions represent the first category addressed in this work and the only one addressed by means of a pattern matching methodology. The rules developed for this module capture some peculiar aspect of this type of definitions, in particular the structure of the part before the connector, in particular, the number of words, the presence of verbal or pronominal forms, and the grammatical category of the first word.

A development corpus, consisting of 75% of the whole corpus, was manually inspected in order to obtain generalizations helping to concisely delimit lexical and syntactic patterns entering in definitory contexts. The held out 25% of the corpus was thus reserved for testing the system.

We end up with a regular grammar based on the tools lxtransduce, a component of the LTXML2 tool set developed at the University of Edinburgh. It is a transducer which adds or rewrites XML markup on the basis of the rules provided.

Initially we developed a baseline grammar for this type of definition. This grammar marked as definition all those sentences containing the colon ":" and the dash "-".

In order to improve this grammar, patterns for punctuation definitions manually marked in the developing corpus were gathered. In order to discover the relevant patterns, all information in these sentences was removed except for the information on part-of-speech. Lists consisting of this information were automatically created in order

to highlight the possible relevant information characterizing the definitory sentences. In this way, it was possible to identify simple rules that are capable to correctly identify punctuation definitions. These rules were applied to the held out corpus in Portuguese and to the entire English corpus.

8.3.3 Copula Definitions

After applying the same rule-based approach to copula definition, we decided that for this category a completely different approach was needed. In particular we chose to deal with this kind of definitions using machine learning classifiers.

The definition extraction problem is envisaged as a binary classification task, where each sentence should be assigned the correct class, i.e. whether it is a definition. In a corpus of naturally occurring texts, it typically happens that the number of sentences expressing a definition is much smaller than the number of sentences that are not definitions. This gives rise to imbalanced datasets that, depending on the corpus, may present different degrees of imbalance, which nevertheless tends to be always quite high. As most of the learning algorithms are designed to maximize accuracy, the imbalance in the distribution of the class tends to lead to a poor performance of these algorithms. For this reason, we sampled the data testing different sampling algorithms.

Regarding the selection of features, other works in the literature using learning algorithms rely strongly on lexical and syntactic components as features to describe the data set, and other types of characteristcs such as the position of the definition inside the document (Joho & Sanderson, 2000), or the presence of determiners in the *definiens* and in the *definiendum* (Fahmi & Bouma, 2006). These features are not only language dependent but also domain dependent, and as we want our methodology to be as general as possible we select the most basic features, that is bi-grams of part of speech (POS). Each sentence was represented as an array whose cells record the number of occurrences of these bi-grams in it.

In order to prepare the data set to be used in our experiments, a simple grammar for each language was created that extracts all the sentences where the verb *to be* appears as the main verb. For Portuguese, we obtained a sub-corpus composed by 1,360 sentences, 121 of which are definitions, with a ratio of about 10:1. For English, the sub-corpus was composed by 2,574 sentences, 40 of which are definitions, with a ratio of 64:1. Finally,

for Dutch we obtained a sub-corpus composed by 4,829, 120 of which were definitions, with a ratio of 39:1.

Five different learning algorithms were used: C4.5, Random Forest, Naïve Bayes, k-NN and SVM. The reason that motivated this choice was twofold: we wanted to cover different classes of algorithms and we wanted to use algorithms representing the state of the art for definition extraction.

Regarding the sampling algorithms, we used two over-sampling algorithms, namely random over-sampling and SMOTE and five under-sampling ones, namely random under-sampling, Condensed Nearest Neighbor Rule (CNN), Edited Nearest Neighbor Rule (ENN), Neighborhood Cleaning Rule (NCL) and Tomek Links.

All the sampling algorithms were first applied one by one and then coupled. In particular, we paired under-sampling algorithms with over-sampling algorithms. In order to assess to which extent a given algorithm is more effective, three different settings were tested: under-sampling to 25%, 50% and 75% and subsequent over-sampling.

8.3.4 Verbal Definitions

For the last type of definitions, we adopted an approach very similar to the one used for copula definitions, that is we built classifiers capable to distinguish between definitory and non-definitory sentences. However, instead of dealing with the problem of data sparseness using sampling algorithms, we modified the dataset, increasing the number of positive examples by using an external source of definitions, namely Wikipedia.

We designed a method to exploit Wikipedia to build a general corpus of definitions that can be used as a starting point in the construction of a definition extractor for verbal definitions. As Wikipedia represents the result of a continuos collaborative effort without a predetermined design, it happens that some knowledge domains are overrepresented, presenting a high number of very specific articles. This characteristic may result in a corpus biased towards some specific domains. When a general corpus of definitions is needed, the way we collect the Wikipedia articles to be used for the construction of this corpus is important.

Exploiting the structure of Wikipedia, more precisely its category structure, we came up with an algorithm for extracting articles over corpora of different sizes, that can be used with different languages. In particular, this algorithm collects all the articles on

the same level of depth in the category tree structure, starting from the *Fundamentals* category.

From the original Portuguese and English corpora, we extracted all the sentences where the connector was represented by a verbal expression contained in a previously gathered list. In this way, we ended up with 3,640 sentences for English, of which 124 were definitions with a ratio of 28:1. For Portuguese, we collected 744 sentences, of which 98 were definitions, with a ratio of 7:1. To these datasets, we added new positive examples in order to obtain new balanced datasets. In this way, 646 new positive examples were added to the Portuguese dataset, while for English, 3392 new positive examples were added.

Furthermore, we used Wikipedia also to extract lexical information to improve the features space for the classifier.

8.4 Reviewing Results

There is no way to directly compare our results to the ones obtained by other studies in the literature, as there is no standard corpus to be used to test a new extractor and because of the difference in the classification of definitions. Nevertheless, it is possible to have a notion of the achievements in this area and place this work among those previously developed by other researchers. Furthermore, there are some results by other researchers we can meaningfully compare with the one presented here as the same corpus of definitions was used.

We offer a summary of the results obtaining for each definition type and we present the global result of the system for the Portuguese (PT), the English (EN) and the Dutch (D) language. We try to compare the performance of our system with state of the art results. In the following tables, results are presented in terms of precision (P), recall (R) and F-measure (F-m).

Table 8.1 shows the best results obtained with the rule based approach for **punctuation definitions**.

When comparing results for punctuation definitions with the other works in this area, there are only three works focussing on punctuation definitions (Borg *et al.*, 2009; Iftene *et al.*, 2008; Westerhout & Monachesi, 2007). Table 8.2 reports the performance

	P	R	F-m
PT	0.87	1	0.93
EN	0.82	1	0.90

Table 8.1: Results for punctuation definitions using rule-based approach for English and Portuguese

	P	R	F-m
Westerhout & Monachesi (2007)	0.77	0.03	0.05
Iftene *et al.* (2008)	0.15	1.0	0.26
Borg *et al.* (2009)	0.33	0.12	0.17

Table 8.2: Results of the state of the art for punctuation definitions

of these works. It is important to highlight that, for English experiments, we used the same corpus as used by Borg *et al.* (2009).

As far as we know, all the three studies have included in punctuation definitions some type of definition we excluded from this category, for instance, parenthetic definitions, as in the example *This leads to what are called electronic spreadsheets (a tool of visualization and simulation used by bookkeepers, economists, and others)*. Even if this kind of sentences contains definitional information, they should form a new type of definitions, called parenthetic definitions, as the structure of the definition is quite different from the regular punctuation definition, this kind of definitional information. The lack of a clear description of this type of definition may have led to the bad performance of the previous systems.

Furthermore, they do not report on special rules for dealing with bibliography sections present in some documents, and they do not mention any kind of lists containing specific words to help to filter non-definitions. These two set of rules also contributed to the performance of our system.

Table 8.3 shows the best results obtained with each different approach for **copula definitions**, that is rule based and machine learning with sampling algorithms. The best performance for the three language was obtained with a Naïve Bayes classifier and when Tomek Links and SMOTE sampling methods were combined together. For Dutch we use the same corpus as used by Westerhout (2009), for this reason our results are directly comparable with theirs in this respect.

	Rule-Based			Sampling		
	P	R	F-m	P	R	F-m
PT	0.32	0.66	0.43	0.95	0.90	0.92
EN	-	-	-	0.92	0.90	0.91
DU	-	-	-	0.98	0.96	0.98

Table 8.3: Results using different approaches with copula definitions for English and Portuguese

	P	R	F-m
Rebeyrolle & Tanguy (2000)	0.17	0.94	0.29
Iftene *et al.* (2008)	0.54	1	0.70
Westerhout (2009)	0.77	0.79	0.78

Table 8.4: Results of the state of the art for copula definitions

Table 8.4 displays results of other works focussing only on copula definitions. The first two are based on pattern matching approach, while the last one uses Balanced Random Forest as a filtering module after the rule-based module. The problem of applying a pattern matching module followed by a filtering module based on a machine learning algorithm is that the classifier can only improve the precision and never the recall. Generally, recall gets worst. And this is what happened with the performance of system reported by Westerhout (2009). The pattern matching obtained a recall of 0.92 and a precision of 0.21 with an F-measure of 0.34. When the learning module was applied, the precision improved greatly and the recall got slightly worst, but the overall performance got better.

Table 8.5 shows the best results obtained with each different approach for **verbal definitions**, that is rule-based, machine learning with sampling algorithms and machine learning with Wikipedia. The best performance for the two languages was obtained with a Naïve Bayes classifier when the datasets were balanced using positive examples automatically extracted from Wikipedia, and when the feature space was improved with lexical information gathered from Wikipedia.

It is interesting to note that there is a significant improvement from an approach to another, from rule-based to sampling and form sampling to Wikipedia. Regarding the last two approaches, the improvement for English is much more evident. This is due to the learning rate with respect to the amount of available data. In the case of

	Rule-Based			Sampling			Wikipedia		
	P	R	F-m	P	R	F-m	P	R	F-m
PT	0.14	0.65	0.23	0.71	0.83	0.77	0.85	0.83	0.84
EN	-	-	-	0.71	0.87	0.78	0.93	0.91	0.92

Table 8.5: Results using different approaches with verbal definitions for English and Portuguese

	P	R	F-m
Sánchez & Márquez (2005)	0.97	1	0.98
Storrer & Wellinghoff (2006)	0.34	0.70	0.46
Iftene *et al.* (2008)	0.76	1	0.86

Table 8.6: Results of the state of the art for verbal definitions

Portuguese, we did not have enough data to obtain a classifier as good as the classifier for English.

Table 8.6 reports on other research focusing on verbal definitions. All three works are based on a pattern matching approach and deal with a very restricted list of verbal patterns. In particular, results presented by Sánchez & Márquez (2005) only refer to the verb *to mean*.

Finally, Table 8.7 shows the overall results of our system. Considering the best results and the proportion of each type of definitions, it is possible to calculate the global performance of the Portuguese and the English definitions extractors.

We can now compare our global result with other general work in the area shown in Table 8.8. Klavans & Muresan (2001), Acosta *et al.* (2011) and Ferneda *et al.* (2012) rely exclusively on a pattern matching approach. Navigli & Velardi (2007) use machine learning (J48 algorithm) to learn the regular expression used by the pattern matching module, while Borg *et al.* (2009) uses genetic algorithms for weighting the manually crafted patterns. Kobyliński & Przepiórkowski (2008) and (Chang & Zheng, 2007)

	P	R	F-m
PT	0.91	0.90	0.90
EN	0.93	0.94	0.93

Table 8.7: Global performance for English and Portuguese definition extractors

	P	R	F-m
Klavans & Muresan (2001)	0.87	0.75	0.80
Chang & Zheng (2007)	0.94	0.75	0.83
Navigli & Velardi (2007)	0.92	0.81	0.86
Kobyliński & Przepiórkowski (2008)	0.21	0.69	0.32
Borg *et al.* (2009)	0.62	0.50	0.57
Acosta *et al.* (2011)	0.57	0.68	0.62
Ferneda *et al.* (2012)	0.76	0.60	0.72

Table 8.8: Results of the state of the art

rely exclusively on machine learning algorithms, in particular the first used a Balanced Random Forest and the second a Support Vector Machine.

8.5 Future Work

As with all research works, the culmination of a research work may represent the starting point of another research path. In our case, one of the most interesting directions to follow could be represented on one hand by a more extensive use of Wikipedia, and on the other hand by a mixture of the present study with the one carried by Navigli & Velardi (2007).

In the first case, concerning Wikipedia, it would be interesting to test the methodology that was used with the verbal definitions also with copula definitions. Instead of modifying the dataset by making use of sampling algorithms, one could increase the number of copula positive examples using new examples extracted from Wikipedia. The opposite would also be interesting to test. In the case of the experiment conducted with the Portuguese language, as it possessed an insufficient number of examples for allowing a good classification, we may apply sampling algorithms to improve the dataset for verbal definitions.

By exploiting Wikipedia, it is also possible to adapt the definitions extractor to a specific domain. Using the same algorithm used to extract a general corpus, it is possible, starting from a specific category, to extract a domain-specific corpus of definitions. These domain specific definitions could be inserted in the dataset instead of the general ones and the lexical and syntactic information gathered from this corpus could be used as features for the classifiers.

In the second case, it could be useful to exploit the approach used in (Navigli & Velardi, 2007), consisting in the automatic discovering of patterns. Then, instead of implementing these patterns in a rule based module, they could be used as features in our system. Again, thanks to Wikipedia, it would be possible to extract general or specific domains patterns in order to fine-tuning a domain specific extractor.

8.6 Final Remarks

The advances reported by this work result from a novel approach to the task of definition extraction.

We obtained a breakthrough in terms of the automatic extraction of definitions by proposing a "divide and conquer" approach. For each definition type, we proposed a different solution where manual effort of generalization is reduced.

For punctuation definitions, we developed a set of rules that may be easily translated into many other languages.

For copula and verbal definitions, we extensively and systematically experimented with different learning algorithms. The major trend in the literature has been to build solutions for this task on the basis of some set of manually crafted patterns. We experimented thoroughly with an alternative solution based solely on machine learning techniques. The key twist to make such an approach not only viable but also with superior results was to focus on the issue of the imbalance of datasets. This permitted to take advantage of the solutions that have been put forward to this problem in recent years, and eventually find out that they allow for a notable breakthrough in terms of the task of definition extraction.

In particular, regarding copula definitions, different sampling techniques and their combination were tested. Thus, the present work also represents a contribution to the seminal work in Natural Language Processing that points towards the importance of exploring the research path of applying sampling techniques to mitigate the bias induced by highly imbalanced datasets, and thus greatly improving the performance of a large range of tools that rely on them.

For verbal definitions a completely different solution was proposed: a new method to increase the number of positive examples was presented based on Wikipedia.

For each definition typology, we based our work on shallow analysis, in particular part of speech information, and external resource that are available for a great number of different languages. This makes the present approach viable for all those languages that are not equipped with rich lexical resources as learning data or in a situation where the domain is too specific to benefit from such resources, and moves away from previous works that use features such as words, word lemmas, position of the sentence in the document, etc.

Furthermore, this research deals with two common problems in NLP research, that is portability and data sparseness. The focus on these two issues represents something new in terms of definition extraction. This approach makes the proposed extractor completely different in nature in comparison to the other works in the field.

Dealing with portability represents something new in the field of definition extraction. A major contribution of this doctoral research work focuses precisely on this issue. The problem here is how to build definitions extractors in a way that their performance is not worse when used in different contexts, where context means a different domain or a different application. The methods proposed here are general enough to be applicable to different languages. This characteristic represents an important innovation in the field of definitions extraction.

References

Acedański, S., Slaski, A. & Przepiórkowski, A. (2012). Machine learning of syntactic attachment from morphosyntactic and semantic co-occurrence statistics. In *Proceedings of the ACL 2012 Joint Workshop on Statistical Parsing and Semantic Processing of Morphologically Rich Languages*, 42–47, Association for Computational Linguistics, Jeju, Republic of Korea. 118

Acosta, O., Sierra, G. & Aguilar, C. (2011). Extraction of definitional contexts using lexical relations. *International Journal of Computer Applications*, 33(6):46–53. 62, 167, 168

Aha, D.W., Kibler, D. & Albert, M.K. (1991). Instance-based learning algorithms. *Machine Learning*, 6(1):37–66. 104

Alarcón, R., Sierra, G. & Bach, C. (2009). ECODE: a definition extraction system. In Z. Vetulani & H. Uszkoreit, eds., *Human Language Technology. Challenges of the Information Society*, 382–391, Springer-Verlag, Berlin, Heidelberg. xiii, 11, 63, 158

Aleven, V., Koedinger, K.R. & Cross, K. (1999). Tutoring answer explanation fosters learning with understanding. In *Proceedings of the 9th International Conference on Artificial Intelligence in Education (AIED'99)*, 199–206, IOS Press. 5

Alshawi, H. (1987). Processing dictionary definitions with phrasal pattern hierarchies. *American Journal of Computational Linguistics*, 13(3-4):195–202. 41, 43

Amsler, R.A. (1981). A taxonomy for english nouns and verbs. In *Proceedings of the 19th Annual Meeting of the Association for Computational Linguistics (ACL'81)*, 133–138. 43

Androutsopoulos, I. & Galanis, D. (2005). A practically unsupervised learning method to identify single-snippet answers to definition questions on the web. In *Proceedings of the Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP'05)*, 323–330, Vancouver, Canada. 41

Aristotle (ca. 350 BCa). *Posterior Analytics*. The Internet Classics Archive. 26

Aristotle (ca. 350 BCb). *Topics*. The Internet Classics Archive. 24, 26

Auger, A. & Barriére, C. (2008). Pattern based approaches to semantic relation extraction: a state-of-the-art. *Terminology*, 14(1):1–19. 43

Aussenac-Gilles, N. & Jacques, M.P. (2006). Designing and evaluating patterns for ontology enrichment from texts. In *Proceedings of the 15th International Conference on Managing Knowledge in a World of Networks (EKAW'06)*, 158–165, Springer-Verlag, Berlin, Heidelberg. 38

Avelãs, M., Branco, A., Del Gaudio, R. & Martins, P. (2008). Supporting e-learning with language technology for Portuguese. In *Proceedings of the International Conference on the Computational Processing of Portuguese (PROPOR'08)*, Springer. 86

Baneyx, A., Malaisé, V., Charlet, J., Zweigenbaum, P. & Bachimont, B. (2005). Synergie entre analyse distributionnelle et patrons lexico-syntaxiques pour la construction d'ontologies différentielles. In *Actes des 6 émes Rencontres Terminologie et Intelligence Artificielle (TIA 2005)*, 31–42. 41

Barnbrook, G. (2002). *Defining Language: a local grammar of definition sentences*. John Benjamins Publishing Company. 1, 20, 21, 25, 43

Batista, G.E.A.P.A., Bazzan, A.L.C. & Monard, M.C. (2003). Balancing training data for automated annotation of keywords: a case study. In *Proceedings of the 2nd Brazilian Workshop on Bioinformatics*, 35–43. 119

Batista, G.E.A.P.A., Prati, R.C. & Monard, M.C. (2004). A study of the behavior of several methods for balancing machine learning training data. *Special Interest Group on Knowledge Discovery and Data Mining Explor. Newsl.*, 6(1):20–29. 93, 119

Batista, G.E.A.P.A., Prati, R.C. & Monard, M.C. (2005). Balancing strategies and class overlapping. In A.F. Famili, J.N. Kok, J.M. Peña, A. Siebes & A.J. Feelders, eds., *Advances in Intelligent Data Analysis VI, 6th International Symposium on Intelligent Data Analysis (IDA'05)*, vol. 3646 of *Lecture Notes in Computer Science*, 24–35, Springer. 101

Bay, S., Kumaraswamy, K., Anderle, M.G., Kumar, R. & Steier, D.M. (2006). Large scale detection of irregularities in accounting data. In *Proceeding of the 6th International Conference on Data Mining*, 75–86, IEEE Computer Society. 91

Biau, G. (2012). Analysis of a random forests model. *Journal of Machine Learning Research*, 13:1063–1095. 118

Blair-Goldensohn, S., McKeown, K. & Schlaikjer, A.H. (2004). Answering definitional questions: A hybrid approach. In *New Directions in Question Answering*, 47–58, AAAI Press. 6, 59

Blitzer, J. (2008). *Domain Adaptation of Natural Language Processing Systems*. Ph.D. thesis, University of Pennsylvania. 9

Borg, C., Rosner, M. & Pace, G. (2009). Evolutionary algorithms for definition extraction. In *Proceedings of the 1st Workshop on Definition Extraction (WDE'09) at Recent Advances in Natural Language Processing (RANLP'09)*, 26–32, Association for Computational Linguistics. 53, 79, 85, 122, 164, 165, 167, 168

Bozzato, L., Ferrari, M. & Trombetta, A. (2008). Building a domain ontology from glossaries: A general methodology. In *Proceedings of the 5th Workshop on Semantic Web Applications and Perspectives (SWAP'08)*, vol. 426. 7

Bradley, A.P. (1997). The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Pattern Recognition*, 30:1145–1159. 105

Branco, A. & Silva, J.R. (2006). LX-Suite: shallow processing tools for Portuguese. In *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL'06)*, 179–183. 161

Breiman, L. (2001). Random Forests. *Machine Learning*, 45(1):5–32. 103

Carrol, J.M. (1995). *Scenario-based design*. John Wiley and Sons, Inc. 149

Chang, C.C. & Lin, C.J. (2001). *LIBSVM: a library for support vector machines*. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm. 104

Chang, X. & Zheng, Q. (2007). Offline definition extraction using machine learning for knowledge-oriented question answering. In *Proceeding of International Conference on Intelligent Computing ICIC (3)*, 1286–1294. 41, 60, 97, 159, 167, 168

Chawla, N.V., Bowyer, K.W., Hall, L.O. & Kegelmeyer, W.P. (2002). Smote: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16:321–357. 102

Chawla, N.V., Japkowicz, N. & Kotcz, A. (2004). Editorial: special issue on learning from imbalanced data sets. *SIGKDD Explorations*, 6(1):1–6. 91

Chen, C., Liaw, A. & Breiman, L. (2004). Using Random Forest to learn imbalanced data. Tech. rep., Department of Statistics, University of Berkeley. 53

Condamines, A. & Rebeyrolle, J. (2001). Searching for and identifying conceptual relationships via a corpus-based approach to a terminological knowledge base. In R. Mitkow, ed., *Recent Advances in Computational Terminology*, 127–148, John Benjamins. 38

Copestake, A. (1993). Defaults in lexical representation. In T. Briscoe, A. Copestake & V. de Paiva, eds., *Inheritance, defaults and the lexicon*, 223–245, Cambridge University Press, New York, NY, USA. 37

Copi, I. & Cohen, C. (1990). *Introduction to logic*. Macmillan. 29, 31, 32, 34

de Freitas, M.C. (2007). *Elaboração automática de ontologias de domínio: discussão e resultados*. Ph.D. thesis, Pontifícia Universidade Católica de Rio de Janeiro. 41

Degórski, Ł., Kobyliński, Ł. & Przepiórkowski, A. (2008a). Definition extraction: improving Balanced Random Forests. In *Proceedings of the International Multiconference on Computer Science and Information Technology (IMCSIT 2008): Computational Linguistics – Applications (CLA'08)*, 353–357, PTI, Wisła, Poland. 53, 118, 119

Degórski, Ł., Marcińczuk, M.M. & Przepiórkowski, A. (2008b). Definition extraction using a sequential combination of baseline grammars and machine learning classifiers. In *Proceedings of the 6th International Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco. 52, 97

Del Gaudio, R. & Branco, A. (2009a). Evaluating a learning management system improved with language technology. In *In proceeding of the 12th International Conference Interactive Computer Aided Learning (ICL'09)*, Villach, Austria. 86, 145

Del Gaudio, R. & Branco, A. (2009b). Improving e-learning experience with language technology: Evaluation results. In *In proceeding of the International Conference Interactive Computer Aided Blended Learning (ICBL'09)*, Frorianopolis, Brazil. 145

Del Gaudio, R., Batista, G. & Branco, A. (2013). Coping with highly imbalanced datasets: A case study with definition extraction in a multilingual setting. *Natural Language Engineering*, FirstView:1–33. 89

Demiröz, G. & Güvenir, H.A. (1997). Classification by voting feature intervals. In *Proceedings of the 9th European Conference on Machine Learning*, 85–92, Springer-Verlag, London, UK. 104

Durán-Muñoz, I. (2010). Specialised lexicographical resources: a survey of translators' needs. In S.G. y M. Paquot, ed., *eLexicography in the 21st century: New Challenges, new applications. Proceedings of ELEX2009*, 7, 55–66, Lovain-La-Neuve. Presses Universitaires de Louvain. 5

Elkan, C. (2001). The foundations of cost-sensitive learning. In *Proceedings of the 17th International Joint Conference on Artificial Intelligence (IJCAI'01)*, 973–978. 92

Fahmi, I. & Bouma, G. (2006). Learning to identify definitions using syntactic feature. In R. Basili & A. Moschitti, eds., *Proceedings of the EACL workshop on Learning Structured Information in Natural Language Applications*, 64–71, Trento, Italy. 60, 97, 99, 129, 159, 162

Fawcett, T. (2004). ROC graphs: notes and practical considerations for researchers. Tech. rep., HP Laboratories. 94

Ferneda, E., do Prado, H.A., Batista, A.H. & Pinheiro, M.S. (2012). Extracting definitions from brazilian legal texts. In B. Murgante, O. Gervasi, S. Misra, N. Nedjah, A.M.A.C. Rocha, D. Taniar & B.O. Apduhan, eds., *Proceedings of the 12th International Conference on Computational Science and Its Applications (ICCSA)*, vol. 7335 of *Lecture Notes in Computer Science*, 631–646, Springer. 50, 167, 168

Friedman, C., Kra, P. & Rzhetsky, A. (2002). Two biomedical sublanguages: a description based on the theories of zellig harris. *Journal of Biomedical Informatics*, 35(4):222–235. 18

Gernot, H. (2007). Defining patterns in translation studies: Revisiting two classics of german translation. *Translationswissenschaft in Target*, 19(2):197–215. 35, 37

Grishman, R., Hirschman, L. & Nhan, N.T. (1986). Discovery procedures for sublanguage selectional patterns: Initial experiments. *Computational Linguistics*, 12:205–215. 20

Grover, C., Matheson, C., Mikheev, A. & Marc, M. (2000). LT TTT - a flexible tokenisation tool. In *In Proceedings of the 2nd International Conference on Language Resources and Evaluation (LREC'00)*. 75

Gupta, A. (2012). Definitions. In E.N. Zalta, ed., *The Stanford Encyclopedia of Philosophy*, Stanford University, fall 2012 edn. 27

Harris, Z. (1968). *Mathematical Structures of Language*. John Wiley & Sons. 18

Hart, P.E. (1968). The Condensed Nearest Neighbor rule. *Information Theory, IEEE Transactions on*, 14(3):515–516. 101

Hearst, M.A. (1992). Automatic acquisition of hyponyms from large text corpora. In *Proceedings of the 14th International Conference on Computational Linguistics (COLING'92*, 539–545, Association for Computational Linguistics, Morristown, NJ, USA. 11, 44, 45, 54, 157

Hurley, P.J. (2011). *A Concise Introduction to Logic*. Clark Baxter. 29, 31

Ide, N. & Suderman, K. (2002). Xml, corpus encoding standard, document xces 0.2. Tech. rep., Department of Computer Science, Vassar College and Equipe Langue et Dialogue, New York, USA and LORIA/CNRS, Vandouvre-les-Nancy, France. 70

Iftene, A., Pistol, I. & Trandabat, D. (2008). Grammar-based automatic extraction of definitions. In *Proceedings of the 10th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing*, 110–115, IEEE Computer Society, Los Alamitos, CA, USA. 52, 79, 164, 165, 166, 167

ISO 1087 (2000). Terminology work: Vocabulary. Tech. rep., ISO. 23, 34

ISO 704 (2009). Terminology work: Principles and methods. Tech. rep., ISO. 22, 34

Jensen, K. & Binot, J.L. (1987). Disambiguating prepositional phrase attachments by using on-line dictionary definitions. *American Journal of Computational Linguistics*, 13(3-4):251–260. 44

John, G.H. & Langley, P. (1995). Estimating continuous distributions in bayesian classifiers. In *11th Conference on Uncertainty in Artificial Intelligence*, 338–345, Morgan Kaufmann. 103

Joho, H. & Sanderson, M. (2000). Retrieving descriptive phrases from large amounts of free text. In *Proceeding of the 9th International Conference on Information and Knowledge Management*, 180–186. 59, 99, 162

Joho, H., Liu, Y.K. & Sanderson, M. (2001). Large scale testing of a descriptive phrase finder. In *Proceedings of the 1st International Conference on Human Language Technology Research (HLT'01)*, 1–3, Association for Computational Linguistics, Morristown, NJ, USA. 58

Klavans, J. & Muresan, S. (2001). Evaluation of the DEFINDER system for fully automatic glossary construction. In *Proceedings of the American Medical Informatics Association Symposium (AMIA'01)*, 324–328. 47, 119, 158, 167, 168

Klavans, J., Popper, S. & Passonneau, B. (2003). Tackling the internet glossary glut: Automatic extraction and evaluation of genus phrases. In *Proceedings of The Special Interest Group on Information Retrieval Workshop on Semantic Web (SIGIR'03)*. 49

Klavans, J.L. & Muresan, S. (2000). DEFINDER: Rule-based methods for the extraction of medical terminology and their associated definitions from on-line text.

In *Proceedings of the American Medical Informatics Association Annual Symposium (AMIA'00)*, 1049. 44, 45, 46

Kobyliński, Ł. & Przepiórkowski, A. (2008). Definition extraction with balanced random forests. In A. Ranta, ed., *International Conference on Natural Language Processing (GoTAL'08)*, 237–247, Springer-Verlag Berlin Heidelberg, Gothenburg. 53, 98, 167, 168

Laurikkala, J. (2001). Improving identification of difficult small classes by balancing class distribution. In *Proceedings of the 8th Conference on AI in Medicine in Europe (AIME '01)*, 63–66, Springer-Verlag, London, UK. 102

Ling, C.X. & Sheng, V.S. (2008). *Cost-Sensitive Learning and the Class Imbalance Problem*, 231–235. Springer. 92

Liu, B., Chin, C.W. & Ng, H.T. (2003). Mining topic-specific concepts and definitions on the web. In *Proceedings of the 12th international conference on World Wide Web (WWW '03)*, 251–260, ACM, New York, NY, USA. 49

Liu, Y., Chawla, N.V., Harper, M.P., Shriberg, E. & Stolcke, A. (2006). A study in machine learning from imbalanced data for sentence boundary detection in speech. *Computer Speech & Language*, 20(4):468–494. 96, 119

Lock, J. (1690). *An Essay Concerning Humane Understanding*. Project Gutenberg. 27

Malaise, V., Zweigenbaum, P. & Bachimont, B. (2004). Detecting semantic relations between terms in definitions. In *the 3rd edition of CompuTerm Workshop (CompuTerm'04) at COLING'04*, 55–62. 9, 11, 46, 54, 55, 158

Markowitz, J., Ahlswede, T. & Evens, M. (1986). Semantically significant patterns in dictionary definitions. In *Proceedings of the 24th Annual Meeting of the Association for Computational Linguistics (ACL'86)*, 112–119. 43

McClosky, D., Charniak, E. & Johnson, M. (2006). Reranking and self-training for parser adaptation. In *Proceedings of the Association for Computational Linguistics (COLING-ACL'06)*, Sydney, Australia. 9

Meyer, I. (2001). Extracting knowledge-rich contexts for terminography. In D. Bourigault, ed., *Recent Advances in Computational Terminology*, 279–302, John Benjamins Publishing. 35, 39, 41, 45

Miliaraki, S. & Androutsopoulos, I. (2004). Learning to identify single-snippet answer to definition questions. In *Proceeding of the 20th International Conference on Computational Linguistics (COLING'04)*, 1360–1366, Geneva, Switzerland. 59, 97, 99, 158

Mill, J.S. (1843). *A System Of Logic, Ratiocinative And Inductive*. Project Gutenberg. 34

Monachesi, P. & Westerhout, E. (2008). What can NLP techniques do for e-learning? In *Proceedings of INFOS 2008*. 146

Monachesi, P., Lemnitzer, L. & Simov, K. (2006). Language technology for elearning. In *Proceedings of European Conference on Technology Enhanced Learning (ECTEL'06)*, Springer. 51, 146

Morin, E. (1999). Automatic acquisition of semantic relations between terms from technical corpora. In TermNet-Verlag, ed., *Proceedings of the 5th International Congress on Terminology and Knowledge Engineering (TKE'99)*, Vienna. 44

Muresan, S. & Klavans, J. (2002). A method for automatically building and evaluating dictionary resources. In *Proceedings of the Language Resources and Evaluation Conference (LREC'02)*, 231–234. 13, 41

Nakamura, J. & Nagao, M. (1988). Extraction of semantic information from an ordinary english dictionary and its evaluation. In *Proceedings of the 12th International Conference on Computational Linguistics (COLING'88)*, 459–464. 41, 43, 44

Nakayama, K., Pei, M., Erdmann, M., Ito, M., Shirakawa, M., Hara, T. & jiro Nishio, S. (2008). Wikipedia mining - Wikipedia as a corpus for knowledge extraction. In *Proceedings of Annual Wikipedia Conference (Wikimania'08)*. 128

Navigli, R. & Velardi, P. (2007). Glossextractor: A web application to automatically create a domain glossary. In *Proceedings of the 10th Congress of the Italian Association for Artificial Intelligence: Artificial Intelligence and Human-Oriented*

*Computing (AI*IA'07)*, 339–349, Springer-Verlag, Berlin, Heidelberg. xiii, 65, 66, 167, 168, 169

Navigli, R. & Velardi, P. (2010). Learning word-class lattices for definition and hypernym extraction. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (ACL'10*, 1318–1327, Association for Computational Linguistics, Stroudsburg, PA, USA. 129, 130

Pantel, P. & Pennacchiotti, M. (2006). Espresso: Leveraging generic patterns for automatically harvesting semantic relations. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics (COLING-ACL'06)*, 113–120, Association for Computational Linguistics, Sydney, Australia. 43, 45

Park, Y., Byrd, R. & Boguraev, B.K. (2002). Automatic glossary extraction: beyond terminology identification. In *Proceeding of the 19th International Conference on Computational Linguistics (COLING'02)*, 1–7. 41

Pearson, J. (1996). The expression of definitions in specialised text: a corpus-based analysis. In M. Gellerstam, J. Jaborg, S.G. Malgren, K. Noren, L. Rogstrom & C. Papmehl, eds., *7th Internation Congress on Lexicography (EURALEX'96)*, 817–824, Goteborg, Sweden. 44, 157

Pearson, J. (1998). *Terms in Context*. John Benjamins Publishing Company. 20, 24, 37, 44, 75, 134, 135

Pinto, A.S. & Oliveira, D. (2004). Extraçcão de definições no Corpógrafo. Tech. rep., Faculdade de Letras da Universidade do Porto. 64

Plato (ca. 360 BC). *Theaetetus*. Project Gutenberg. 1, 26

Prager, J., Radev, D. & Czuba, K. (2001). Answering what-is questions by virtual annotation. In *Proceedings of the 1st International Conference on Human Language Technology Research (HLT'01)*, 1–5, Association for Computational Linguistics, Morristown, NJ, USA. 44, 57, 58, 59

Prati, R.C., Batista, G.E.A.P.A. & Monard, M.C. (2011). A survey on graphical methods for classification predictive performance evaluation. *IEEE Transactions on Knowledge and Data Engineering*, 23(11):1601–1618. 93

Przepiórkowski, A., Degórski, L., Spousta, M., Simov, K., Osenova, P., Lemnitzer, L., Kubon, V. & Wójtowicz, B. (2007a). Towards the automatic extraction of definitions in Slavic. In J. Piskorski, B. Pouliquen, R. Steinberger & H. Tanev, eds., *Proceedings of the Workshop on Balto-Slavonic Natural Language Processing (ACL'07)*, 43–50, Association for Computational Linguistics. 51

Przepiórkowski, A., Degórski, L. & Wójtowicz, B. (2007b). On the evaluation of polish definition extraction grammars. In Z. Vetulani, ed., *Proceedings of the 3rd Language and Technology Conference*, 473–477. 52

Przepiórkowski, A., Marcińczuk, M. & Degórski, Ł. (2008). Noisy and imbalanced data: Machine learning or manual grammars? In *9th International Conference on Text, Speech and Dialogue (TSD'08)*, 169–176, Lecture Notes in Artificial Intelligence, Berlin, Springer-Verlag, Brno, Czech Republic. 118

Quinlan, J.R. (1996). Improved use of continuous attributes in c4.5. *Journal of Artificial Intelligence Research*, 4:77–90. 103

Rebeyrolle, J. (2000). Utilisation de contextes définitoires pour l'acquisition de connaissances à partir de textes. In P. Tchounikine, ed., *Actes Journées Francophones d'Ingénierie de la Connaissance (IC'00)*. 61

Rebeyrolle, J. & Tanguy, L. (2000). Repérage automatique de structures linguistiques en corpus : le cas des énoncés définitoires. In *Cahiers de Grammaire*, 140–150. xiii, 61, 62, 64, 166

Robinson, R. (1950). *Definition*. Oxford University Press. xi, 25, 28, 33

Roth, D. (1999). Learning in natural language. In *Proceedings of the 16th International Joint Conference on Artificial intelligence (IJCAI'99)*, vol. 2, 898–904, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA. 118

Sager, N., Lyman, M., Bucknall, C., Nhan, N. & Tick, L.J. (1994). Natural language processing and the representation of clinical data. *Journal of the American Medical Informatics Association*, 1(2). 18

Saggion, H. (2004). Identifying definitions in text collections for question answering. In *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC'04)*, European Language Resources Association. 41, 58, 99

Sánchez, A. & Márquez, M. (2005). Hacia un sistema de extracción de definiciones en textos jurídicos. In *Actas de la 1er Jornada Venezolana de Investigación en Lingüística e Informática.*, 1–10, Venezuela. 13, 45, 50, 167

Sanger, J.C. & Ndi-Kimbi, A. (1995). The conceptual structure of terminological definitions and their linguistic realisations. *Terminology*, 2(1):61–85. 37

Seppälä, S. (2009). A proposal for a framework to evaluate feature relevance for terminographic definitions. In *Proceedings of the 1st Workshop on Definition Extraction (WDE'09) at Recent Advances in Natural Language Processing (RANLP'09)*, 47–53, Association for Computational Linguistics. 3, 41

Sierra, G., Alarcón, R., Aguilar, C. & Barrón, A. (2006). Towards the building of a corpus of definitional contexts. In *Proceeding of the 12th EURALEX International Congress*, 229–240. xi, 3, 36, 38

Silva, J.R. (2007). *Shallow Processing of Portuguese: From Sentence Chunking to Nominal Lemmatization*. Master's thesis, Universidade de Lisboa, Faculdade de Ciências. 71

Spinoza, B. (1677). *Ethics*. Project Gutenberg. 26

Storrer, A. & Wellinghoff, S. (2006). Automated detection and annotation of term definitions in German text corpora. In *Proceedings of International Conference on Language Resources and Evaluation (LREC'06)*. 7, 54, 56, 167

Symonenko, S., Rowe, S. & Liddy, E.D. (2006). Illuminating trouble tickets with sublanguage theory. In *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers (NAACL-Short '06)*, 169–172, Association for Computational Linguistics, Stroudsburg, PA, USA. 19

Taft, L.M., Evans, R.S., Shyu, C.R., Egger, M.J., Chawla, N., Mitchell, J.A., Thornton, S.N., Bray, B. & Varner, M. (2009). Countering imbalanced datasets to improve adverse drug event predictive models in labor and delivery. *Journal of Biomedical Informatics*, 42:356–364. 91

Tjong, E., Sang, K., Bouma, G. & de Rijke, M. (2005). Developing offline strategies for answering medical questions. In *Proceedings of the AAAI-05 Workshop on Question Answering in Restricted Domains*, 41–45. 41, 58, 89

Tomanek, K. & Hahn, U. (2009). Reducing class imbalance during active learning for named entity annotation. In *Proceedings of the 5th International Conference on Knowledge Capture (K-CAP'09)*, 105–112, ACM, New York, NY, USA. 95

Tomek, I. (1976). Two modifications of CNN. *Systems, Man and Cybernetics, IEEE Transactions on*, 6(11):769–772. 101

Tomuro, N. & Shepitsen, A. (2009). Construction of disambiguated folksonomy ontologies using Wikipedia. In *Proceedings of the Workshop on The People's Web Meets NLP (People's Web '09)*, 42–50, Association for Computational Linguistics, Morristown, NJ, USA. 128

Toutanova, K. & Manning, C.D. (2000). Enriching the knowledge sources used in a maximum entropy part-of-speech tagger. In *Proceedings of the International Conference on Empirical Methods in Natural Language Processing and Very Large Corpora (EMNLP'00)*, vol. 13, 63–70, Association for Computational Linguistics, Stroudsburg, PA, USA. 71, 161

Vatturi, P. & Wong, W.K. (2009). Category detection using hierarchical mean shift. In *Proceedings of the 15th International Conference on Knowledge Discovery and Data Mining (KDD'09)*, 847–856, ACM, New York, NY, USA. 91

Vossen, P. & Copestake, A. (1993). Untangling definition structure into knowledge representation. In T. Briscoe, A. Copestake & V. de Paiva, eds., *Inheritance, defaults and the lexicon*, 246–274, Cambridge University Press, New York, NY, USA. 44

Walter, S. & Pinkal, M. (2006). Automatic extraction of definitions from german court decisions. In *Proceedings of the Workshop on Information Extraction Beyond*

The Document, 20–28, Association for Computational Linguistics, Sydney, Australia. 7, 13, 41, 54, 55

Weiss, G., McCarthy, K. & Zabar, B. (2007). Cost-sensitive learning vs. sampling: which is best for handling unbalanced classes with unequal error costs? In R. Stahlbock, S.F. Crone & S. Lessmann, eds., *Proceedings of the International Conference on Data Mining*, 35–41, CSREA Press. 92

Weiten, W., Deguara, D., Rehmke, E. & Sewell, L. (1999). University, community college, and high school students' evaluations of textbook pedagogical aids. *Teaching of Psychology*, 26:19–21. 5, 146

Westerhout, E. (2009). Extraction of definitions using grammar-enhanced machine learning. In *Proceedings of the Student Research Workshop at the Conference of the European Chapter of the Association for Computational Linguistics (EACL'09)*, 88–96, Association for Computational Linguistics, Athens, Greece. 53, 98, 118, 119, 165, 166

Westerhout, E. & Monachesi, P. (2007). Extraction of dutch definitory contexts for elearning purposes. In *Proceedings of the Computational Linguistics in the Netherlands (CLIN'07)*, 219–234. 12, 53, 79, 85, 98, 118, 160, 161, 164, 165

Westerhout, E. & Monachesi, P. (2008). Creating glossaries using pattern-based and machine learning techniques. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC'08)*, 3074–3081. 53, 97, 122, 158

Wilson, D.L. (1972). Asymptotic properties of nearest neighbor rules using edited data. *IEEE Transactions on Systems, Man, and Cybernetics*, 2:408–421. 101

Witten, I.H. & Frank, E. (2005). *Data Mining: pratical machine learning tools and techniques (Second Edition)*. Morgan Kaufmann. 65, 100, 104

Wu, G. & Chang, E. (2003). Class-boundary alignment for imbalanced dataset learning. In *Proceedings of the 20th International Conference on Machine Learning (ICML 2003) - Workshop on Learning from Imbalanced Data Sets*, 786–795. 91

Wüster, E., Cabré, M. & Nokerman, A. (1998). *Introducción a la teoría general de la terminología y a la lexicografía terminológica*. Sèrie Monografies, Institut Universitari de Lingüística Aplicada, Universitat Pompeu Fabra. 37

Zanzotto, F.M. & Pennacchiotti, M. (2010). Expanding textual entailment corpora from wikipedia using co-training. In *Proceedings of the COLING-Workshop on The People's Web Meets NLP: Collaboratively Constructed Semantic Resources*. 128

Zesch, T. & Gurevych, I. (2007). Analysis of the Wikipedia Category Graph for NLP Applications. In *Proceedings of the TextGraphs-2 Workshop (NAACL-HLT)*. 128

Zesch, T., Müller, C. & Gurevych, I. (2008). Extracting lexical semantic knowledge from Wikipedia and Wiktionary. In N. Calzolari, K. Choukri, B. Maegaard, J.O. Joseph Mariani, S. Piperidis & D. Tapias, eds., *Proceedings of the 6th International Language Resources and Evaluation (LREC'08)*, European Language Resources Association (ELRA), Marrakech, Morocco. 128, 131

Zhang, H. (2005). Exploring conditions for the optimality of Naïve Bayes. *International Journal of Pattern Recognition and Artificial Intelligence (IJPRAI'05)*, 19(2):183–198. 118

Zhu, J. (2007). Active learning for word sense disambiguation with methods for addressing the class imbalance problem. In *Proceeding Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, 783–790. 95