Definition extraction for glossary creation

A study on extracting definitions
for semi-automatic glossary creation in Dutch

Published by
LOT
Janskerkhof 13
3512 BL Utrecht
The Netherlands

Phone: +31 30 253 6006
Fax: +31 30 253 6000
e-mail: lot@uu.nl
http://www.lotschool.nl/

Cover illustration: Pieter Bruegel the Elder, *The Tower of Babel*, oil on panel, c. 1563, Kunsthistorisches Museum Vienna.

ISBN 978-94-6093-034-8
NUR 616

Definition extraction for glossary creation

A study on extracting definitions for semi-automatic glossary creation in Dutch

Definitie-extractie voor het creëren van glossariums

Een onderzoek naar de extractie van definities voor het semi-automatisch creëren van Nederlandse glossariums

(met een samenvatting in het Nederlands)

Proefschrift

ter verkrijging van de graad van doctor
aan de Universiteit Utrecht
op gezag van de rector magnificus, prof.dr. J.C. Stoof,
ingevolge het besluit van het college voor promoties
in het openbaar te verdedigen op
vrijdag 2 juli 2010 des morgens te 12.45 uur

door

Elizabeth Nicoline Westerhout

geboren op 21 september 1983 te Lienden

Promotor:	Prof.dr. J.E.J.M Odijk
Co-promotor:	Dr. P. Monachesi

The beginning of wisdom is the definition of terms

(Socrates)

Contents

Acknowledgements

Writing a dissertation often means that you are working alone and sometimes you can get the feeling 'I'm a poor lonesome researcher'[1]. Luckily, there were many people who accompanied me during my research journey, which made my research and writing process less lonesome. It's the invaluable support of these people that made it possible to write my dissertation.

First and foremost I would like to thank my supervisor and co-promotor Paola Monachesi. I probably would never have written a PhD thesis without her support and guidance. She supervised my Master's thesis and asked me to join the Language Technology for eLearning (LT4eL) project as a researcher in December 2005. I always liked doing research, but I never thought of writing a PhD thesis. During the project, I became more and more interested in the topic of definition extraction for glossary creation. I am grateful to UiL OTS for giving me the opportunity to work out my ideas and experiments as a PhD student. I am also indebted to my promotor Jan Odijk for his contributions, insights and detailed comments.

I would like to thank all the members of the LT4eL project. A special thanks to Lothar Lemnitzer, who guided the definition extraction work within the project, and to Claudia Borg, Rosa Del Gaudio, and Lukasz Degórski from the machine learning group. It always has been inspiring and great to work with you! I am grateful to Eelco Mossel, Miroslav Spousta, and Lukasz Kobyliński for their help in writing the scripts that were needed for conducting the experiments.

Nicole Grégoire, Eelco Mossel, Claudia Borg and Marieke Westerhout, thank you very much for reading (parts of) my thesis and giving suggestions for improving the content, style and structure of the thesis.

The last people I would like to thank are my family and friends. You were always willing to offer distraction and moral support when I needed it. Taking the time to relax and to do completely 'thesis-unrelated' things was essential and you pushed (or sometimes even forced) me to do this regularly.

[1] Adapted version of Lucky Lukes quote ('I'm a poor lonesome cowboy')

Plato defined man thus: "Man is a two-footed, featherless animal;" and was much praised for the definition; so Diogenes plucked a cock and brought it into his school, and said, "This is Plato's man."

Laertius

1 Introduction

It happens to everyone: you are reading a book and you encounter a word you do not know. Generally, you try to infer the meaning of the word from the context it is in, but if this is not enough you must then turn to a dictionary or glossary if available. Glossaries are commonly found in technical literature, manuals and textbooks. They provide definitions for a list of terms that are discussed in the book. In an online learning environment, teachers provide digital learning materials to their students. More often than not, these documents do not contain a glossary. Since teachers do not have the time to create them, the learner needs to search for the definitions of the important terms himself. Some definitions may be mentioned somewhere in the book, but it would take the learner a lot of time and effort to locate them. An alternative and probably quicker solution would be to search for the definition in an external source, such as an encyclopaedia. In this case, some efforts are required from the learner as well. An additional problem is that for many terms there exists more than one definition. The learner has to be able to select the correct explanation that matches the meaning intended in the text.

A method to automatically retrieve definitions from texts is of great value to the learner. He can use it to compile a list of definitions on the basis of the text he has to study. This assures him that the definitions capture the correct meaning of the terms and it enables him to spend more time to the actual learning process. To create such a tool, it is necessary to understand first what constitutes a good definition. Humans are very good at distinguishing definitions from other sentences. Even when they do not understand the content of the definition, they can

nevertheless judge whether or not a sentence is a definition. The challenge we are facing is to develop a method that is able to distinguish definitions from non-definitions automatically.

This thesis presents research on the automatic extraction of definitions on the basis of electronic texts. The focus is on one of the applications in which definitions are relevant – the glossary creation context. The Language Technology for eLearning project (LT4eL)[1] has been the starting point of our research. The aim of this project was to develop solutions that facilitate the retrieval of documents in an online learning environment through the use of language technologies and semantic knowledge. One of the tasks within the project involved the development of a tool to create glossaries on the basis of documents semi-automatically. This glossary creation tool can be embedded in an online learning environment to offer tutors and learners the possibility to semi-automatically compile glossaries for their learning materials.

A sequential combination of a pattern-based approach and machine learning techniques has been used to extract definitions. In the pattern-based step, each sentence is compared against a set of patterns that are common to definitions to extract all sentences that have a definition pattern. Since many definition patterns can be used in non-definitions, the pattern-based approach returns a number of non-definitions as well. Machine learning techniques based on several types of information are employed to filter these non-definitions. With this combined approach, the majority of definitions in each text can be found. Although there are still some non-definitions left, the number of extracted non-definitions has been reduced considerably by the machine learning classifier.

1.1 DEFINITIONS

The Longman dictionary defines the word 'definition' as "A phrase or sentence that says exactly what a word, phrase, or idea means". In some contexts, such as the dictionary context, it is indeed important that this is the case and that accurate definitions are used. The example on Plato and Diogenes at the beginning of this chapter illustrates that misunderstandings can arise when a definition does not exactly

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

describe the meaning of a term. However, there are also situations that do not require a phrase that 'says exactly' what something means and in such cases a weaker version of definition suffices: "A phrase or sentence that makes clear what a word, phrase, or idea means". It is thus not necessary that the exact meaning is given, but it is enough to provide a description of how to employ the word, e.g. by providing a description of its function. To distinguish between the two types of definitions, the phrases that fit the Longman dictionary description are called 'narrow definitions' whereas the others are referred to as 'broad definitions'. The distinction between broad and narrow definitions is not absolute. Example 1 shows three possible definitions for the term 'HTML' in which the level of detail differs. Example 1a is a narrow definition that provides detailed information on the term whereas example 1c is a broad definition which gives only the class to which the term belongs. In example 1b, the level of detail is in between the two others – the definition is less detailed than the first one, but more specific than the last one.

(1) a. HyperText Markup Language (afgekort HTML) is een
HyperText Markup Language (for short HTML) is a
opmaaktaal voor de specificatie van documenten
markup language for the specification of documents
op het World Wide Web ontwikkeld door Tim Berners Lee
on the World Wide Web developed by Tim Berners Lee
in 1990 .
in 1990 .

'*Hypertext Markup Language (HTML for short) is a markup language for the specification of documents on the World Wide Web developed by Tim Berners Lee in 1990.*'

b. HyperText Markup Language (afgekort HTML) is een
HyperText Markup Language (for short HTML) is a
opmaaktaal voor de specificatie van documenten
markup language for the specification of documents
op het World Wide Web .
on the World Wide Web .

'*Hypertext Markup Language (HTML for short) is a markup language for the specification of documents on the World Wide Web.*'

c. HyperText Markup Language is een opmaaktaal .
HyperText Markup Language is a markup language .
'Hypertext Markup Language is a markup language.'

Depending on the context and situation it may sometimes be necessary to use narrow definitions whereas at other times both broad and narrow definitions can be employed. Narrow definitions clearly have to be used in the dictionary context, since the purpose of dictionaries is to provide users with an accurate, non-ambiguous description of an unknown term. In this context, broad definitions may cause misunderstandings, which is exactly something a dictionary strives to prevent. A context in which a narrow view on definitions may not be necessary is the glossary context on which our research focuses. The main purpose of the glossary definitions is to provide the reader with an explanation of the most important terms. Broad definitions will often suffice in this area, since the learner may already have some knowledge about the domain. As a consequence, definitions do not necessarily have to be narrow definitions, since broad definitions often provide enough information. In addition to the dictionary and the glossary context, two other areas in which definitions are relevant are question answering and ontology building. Section 1.4 compares the glossary creation domain to the three other applications.

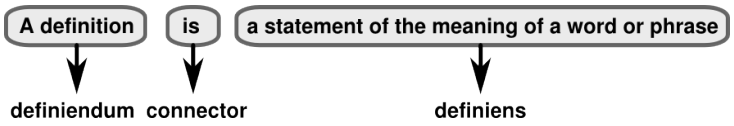

Figure 1.1 *Elements of a definition*

Definitions can be grouped in different ways. The classification methodology adopted within this thesis is a pattern-based one (Westerhout and Monachesi, 2007a). It is based on the assumption that both broad and narrow definitions consist of (at least) three elements, namely a definiendum, definiens and connector (Walter and Pinkal, 2006). Figure 1.1 shows an example definition in which these three constituents are present. The definiendum, often the subject, is the element that is defined (Latin: *that which is to be defined*). The definiens provides the meaning of the definiendum (Latin: *that which is doing the defining*). The

definiendum and the definiens are linked via the connector, which can be a verbal phrase or a punctuation character. The connector indicates what the relation is between definiendum and definiens.

Type	Example
is	Gnuplot is a program for drawing graphs.
verb	A topic map consists of associated topics and represents knowledge.
punctuation	SMS messages: short text messages from up to 160 characters.
pronoun	Dedicated readers. These are special devices, developed to make it possible to read e-books.

Table 1.1 *Examples for each of the definition types*

Westerhout and Monachesi (2007a) proposed four types of definitions on the basis of the three elements (Table 1.1). Three of them can be characterized on the basis of their connector pattern. The first and most common connector is a form of the auxiliary verb 'to be'. The second type of connector includes all other verbal phrases that are employed in definitions, such as 'mean' and 'consist of'. Punctuation characters like colons and brackets constitute the third group of connectors. The last type is different from the others, since in this type a pronoun - sometimes in combination with a connector - is the distinctive feature. The pronoun refers to a definiendum or definiens in a previous (part of the) sentence.

1.2 THE LANGUAGE TECHNOLOGY FOR ELEARNING PROJECT

The glossary candidate detector has been developed for an eLearning environment. The term eLearning is the collective term for modeling formal and informal learning situations using information and communication technology, especially Internet technology. Integrating language technology techniques and tools in eLearning applications can enhance the quality and speed of the learning process. Since such tools are intended to improve the learning experience of the learners, they

should offer learners added value from a pedagogical perspective. In addition, they have to be easy and intuitive to use and should speed up the learning process.

The development of the Dutch glossary candidate detector that is presented in this thesis has been initiated within the Language Technology for eLearning (LT4eL) project (2005-2008)[2]. The aim of this European project was to develop multilingual language technology techniques and tools to be integrated into eLearning applications (Monachesi et al., 2006). The languages represented in the project are Bulgarian, Czech, Dutch, German, English, Polish, Portuguese, and Romanian. In addition to the glossary candidate detector, within LT4eL a keyword extractor, a domain ontology, and semantic search on the basis of this ontology have been developed. The LT4eL tools and ontology have been integrated in a Learning Management System (LMS). An LMS is an important application within the eLearning domain used for delivering, tracking and managing learning activities. Most LMSs are web-based to facilitate access to learning content and administration. For evaluation and validation purposes, the LT4eL tools and resources have been integrated into the LMS ILIAS[3].

	#
Documents	45
Sentences	31552
Words	420202
Definition sentences	709

Table 1.2 *Corpus statistics for the Dutch LT4eL corpus*

Within the LT4eL project, a corpus has been collected and annotated for the development of the definition extractor, keyword detector, and ontology in each of the eight LT4eL languages. The documents of the corpus are all learning objects. Furthermore, from the large variety of possible learning objects that are available on the web (e.g. manuals, videos, slides, lecture notes), only textual documents have been selected for the LT4eL corpus since the aim of the LT4eL project was to

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

[3]http://www.ilias.de/

apply natural language processing techniques on the documents. The Dutch LT4eL corpus consists of 45 text documents in the domains of computing and eLearning. It contains descriptive documents on computing and eLearning, manuals on computer programs and systems, such as Word, LaTeX, and Unix and documents introducing academic skills to students, such as preparing a presentation or writing a paper. Table 1.2 provides some general statistics of the Dutch LT4eL corpus.

The definition extractor implemented in the glossary candidate detector has been developed and evaluated on the basis of this corpus. The definitions from the corpus have been investigated to gain insights into characteristics of the ways in which definitions are used in learning objects. The first version of the glossary detector presented in this thesis has been developed within the context of the LT4eL project and is based on the patterns observed in the manually selected definitions.

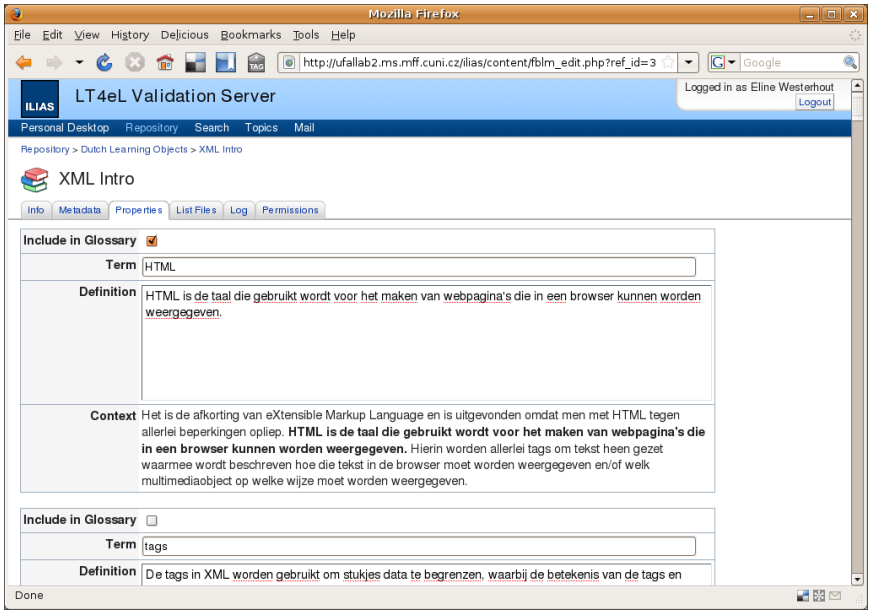

Figure 1.2 *A screenshot of the integrated glossary candidate detector within ILIAS*

Figure 1.2 shows a screenshot of the LT4eL glossary candidate detector integrated within the LMS ILIAS. The tool takes as input a learning object (TXT, PDF, DOC, or HTML) and offers as output a list of definition candidates. The candidates are surrounded by two context

sentences, namely one sentence to the left and one sentence to the right. The user can select the correct definitions from the list and adjust or extend them on the basis of the left and right context information and his or her own knowledge.

1.3 DEFINITIONS IN ELEARNING

Glossaries can play an important role within eLearning since they support a learner in understanding the learning object he is confronted with by explaining the central concepts which are being conveyed in it. In this way, the glossaries can enhance the learning process of learners, since they do not have to search for the definitions themselves. A possible way to create glossaries for learning objects would be linking existing glossaries to them. However, an obvious shortcoming of this approach is that the learner could be confronted with many definitions for a term he is looking for instead of only the definitions that are appropriate in the given context. Another disadvantage of this option is that very detailed glossaries would be necessary to ensure that all relevant terms of each randomly selected document within the domain are defined in it. Moreover, in several domains, such as the computing domain, new terms are often introduced which will be not directly available in existing dictionaries or glossaries.

An alternative – that has been adopted in our research – is to build glossaries on the basis of the learning objects themselves. By doing so, the exact definition as employed by the author of a certain document is captured; this definition often deviates from a more general definition of the term or a definition of the same term in another domain or context. The learning process is thus facilitated by providing the most appropriate definition to the learner for the concept he is not familiar with. In addition, the glossary gives the learner a quick impression of the content of the learning object, which can serve as a summary. On the basis of this summary, he cannot only learn the most important terms, but also decide whether or not the document is relevant for him.

In addition to searching for narrow definitions, the purpose within the glossary creation context is to detect broad definitions as well. They can contain, for example, information on how a concept should be used

or what its purpose is. In the context of glossary creation, where the glossary has to provide learners with the meaning of unknown concepts to improve their learning process, broad definitions are useful as well, since they generally provide the learner with enough information to understand a term.

A second consequence of applying definition extraction within the domain of glossary creation in eLearning is that it is important to obtain a high recall. A high recall means that most of the definitions that are present in a text are detected. The glossary is based on a learning object and should contain definitions for its key terms. For the terms for which no definition is detected automatically, the learner has to find or formulate them himself. Clearly, this leads to an increased workload for the learner.

The inevitable countereffect of assigning more importance to obtaining a high recall is that the precision will decrease. The precision indicates which proportion of the retrieved sentences are indeed definitions. The goal is to find a good balance between on the one hand finding as many relevant definitions as possible (high recall) and on the other hand keeping the number of extracted non-definitions as low as possible (precision not too low).

1.4 DEFINITIONS IN OTHER APPLICATIONS

Next to glossary creation, Section 1.1 mentioned three other possible applications in which definitions play a role. These are question answering (QA), dictionary building and ontology building. This section compares the three applications with the glossary creation application.

Glossary creation versus question answering The most important difference between glossary creation and QA is that in QA the definiendum is known in advance, since it is already included in the question (cf. Voorhees (2002); Joho and Sanderson (2000); Saggion (2004)). A question analysis component is generally employed in QA to limit the number of definition candidates to a set of sentences that contain this definiendum before the definitions are extracted. For example, when the question 'What is XML?' has to be answered, the set of possible an-

swers can be restricted to sentences containing the term XML or one of its synonyms. In the glossary creation context, the terms that will be included in the glossary are not known in advance. As a consequence, it is not possible to start with the definiendum. Instead, the extraction has to start with the characteristics inherent to definitions itself, such as the lexico-syntactic patterns used: any pattern that can be used in a definition could be relevant and needs to be selected.

Another practical difference between QA and glossary creation is that in the glossary creation context one does not have to select the best answer(s) to a certain question, like in QA (cf. Joho and Sanderson (2000); Han et al. (2007)), but have to decide for each individual sentence whether or not it can be a definition. The user can either select the best one or combine different definitions into one definition that covers all important aspects of the term. In fact, more definitions for the same term in the set of extracted sentences can be quite useful for the creation of a glossary, since it is very well possible that a combination of different definitions provides more insights on a term.

Glossary creation versus dictionary building Definition extraction for dictionary building is more closely related to the glossary creation context (Muresan and Klavans, 2002). Just as in glossary building, the terms that should come in the dictionary are not always known beforehand. At first sight, definition extraction for glossary creation within eLearning and definition extraction for dictionary building might even seem to be the same task. However, there are at least two differences between the tasks.

The first difference is related to the corpus that is used. In the glossary creation context, the 'corpus' from which the information has to be extracted consists of only one document and as many definitions as possible should be detected within this text. In dictionary building, a large corpus can be used, since the aim is to find the best definition for each term. The consequence is that in glossary creation it is more crucial to obtain a high recall. It is in this application not enough to include only the most common definition patterns, that is, the definitions in which a form of the verb 'to be' is used as connector. Since in dictionary building a larger corpus can be considered to find definitions, it is

acceptable to focus only on such common definitions.

A second difference is that in a glossary it is possible that a definition is included which is not the commonly accepted definition of a word, but only a meaning attached to it within that specific text. In the glossary, the description used by the author is important whereas a dictionary definition should express a common meaning of a word.

The creation of specialized dictionaries is more similar to our context. In this case, only the first restriction applies, since it is still possible to use a larger corpus for the creation of the ontology. Just like in the glossary creation context, the extraction method should be able to differentiate between ambiguous terms and to select the appropriate definition in the specialized dictionaries as well.

Glossary creation versus ontology building Definitions can be used in two ways in the ontology building context. The first application of definitions is to describe concepts from an ontology. One can either start from the definiendum (concept) or from the pattern of the definition for the extraction. If the first option is selected, the extraction procedure is similar to the QA context, where the term is known in advance as well. The alternative is to start from the definition pattern. In this case, the situation is similar to the glossary creation context. The main difference with the glossary creation context in this respect is that definitions for ontologies need to be narrow, since they have to provide the exact meaning of the term.

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 and Wellinghof, 2006; Walter and Pinkal, 2006). A semantic relation expresses how two terms or phrases are related to each other. Often, this semantic relation is expressed by the verbal phrase that connects two parts. 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 enrich ontologies automatically. Both new concepts can be added and new relations can be expressed on the basis of the definitions. However, although the number

of semantic relations is higher in the glossary creation context, there are 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.

1.5 THE EXTRACTION OF DEFINITIONS

For each of the four different applications in which definitions play a role, research has been carried out to detect them automatically. Different approaches have been investigated to find out which method produces the best results. Generally, these approaches are independent of the context in which they are employed, although the exact way in which they are used can differ depending on the application.

Initially, automatic definition extraction has been mainly addressed using pattern-based approaches. These approaches are often based on patterns represented by regular expressions that are common in definitions (Joho and Sanderson, 2000; Prager et al., 2002) or on a more complex representation of the lexico-syntactic patterns of sentences (Muresan and Klavans, 2002; Saggion, 2004; Storrer and Wellinghof, 2006; Walter and Pinkal, 2006). More recently, the pattern-based approaches have been complemented with or completely replaced by machine learning approaches (Blair-Goldensohn et al., 2004; Miliaraki and Androutsopoulos, 2004; Fahmi and Bouma, 2006).

In our research – which started in 2005 – initially a pattern-based approach inspired by the work from Muresan and Klavans (2002) has been adopted (Westerhout and Monachesi, 2007a). During the development of the pattern-based approach, a number of shortcomings emerged that are difficult to address with the pattern-based approach itself. The most important problem is the fact that definition patterns are often used in non-definitions as well. To solve this problem, a machine learning approach is used in succession to the pattern-based approach as a filtering or refining step (Westerhout and Monachesi, 2007b, 2008; Westerhout, 2009a,b). The advantage of applying machine learning techniques is that the system can be employed to investigate for a set of manually selected features which of them are best at distinguishing definitions from non-definitions. It has been investigated in this

thesis which combinations of features perform best. Figure 1.3 shows the steps involved in the process of extracting a glossary from a document that have been followed in this thesis. The purely pattern-based approach is presented in Chapter 3 and the pattern-based approach enhanced with machine learning is described in Chapter 4 and 5.

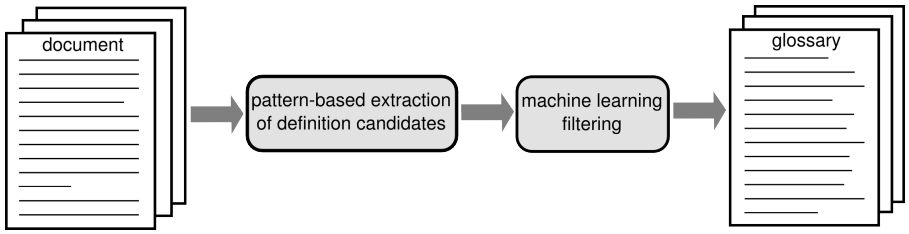

Figure 1.3 *From document to glossary*

When applied to the domain of glossary creation for eLearning, the current approaches for definition extraction suffer from a number of shortcomings which are addressed in this thesis. A first shortcoming is a side-effect from the fact that most definition extraction research has been carried out in the context of question answering. For this application, the identification of only a limited number of definition types is often enough. As a consequence, it has not been necessary to use a more extended typology. The typology employed in this thesis considers definition types that are observed less frequently as well, such as punctuation and pronoun definitions.

A second disadvantage of existing approaches is that most of them extract definitions from very structured texts, such as Wikipedia. They have not been designed for dealing with randomly selected learning objects, which are generally less structured than Wikipedia articles. The LT4eL corpus employed for developing and testing our application contains texts of different lengths (1-100 pages), styles (e.g. descriptive, manuals), levels (from beginners to experts), and genres. As all texts have been collected from the web, they give a good impression of what can be expected from web documents.

The third shortcoming is that it has not been investigated in detail which characteristics of definitions perform best on the classification task. This thesis presents experiments that implement a large variety

of definition characteristics (e.g. linguistic, position, layout) and investigates which are the most suitable ones for the classification of definitions.

1.6 RESULTS

As explained in the previous section, the extraction approach adopted in this thesis consists of two steps. In the first step, the lexico-syntactic patterns of definitions have been used to extract all sentences having a definition pattern from the LT4eL corpus. Not only *is* definitions, but also other types of definitions are addressed. The results are evaluated for each of the four definition types individually. The evaluation shows that it is possible to obtain a high recall with this approach. It is more difficult to get a high precision, since the definition patterns are often used in non-definitions as well. Especially the *punctuation* and *pronoun* definitions turn out to be problematic in this respect. Although the precision scores for the *is* and *verb* definitions are better than the *punctuation* and *pronoun*, also for these types there is room for improvement.

In order to improve the low precision scores obtained with the pattern-based approach, machine learning classifiers have been trained. Since the recall obtained with the pattern-based method is acceptable, the sentences extracted with this approach have been used as input. The advantage of using these data is that a large number of non-definitions has been filtered out already. In the machine learning experiments, six types of features have been distinguished to filter out non-definitions. These features include linguistic *n*-grams, linguistic properties of definiendum and definiens, information on the connector, position, layout, and importance of the defined terms (referred to as 'keyword information'). The best results are generally obtained with the linguistic, connector and position information. The layout and keyword information cannot be used individually to predict whether or not a sentence is a definition. When the different settings are combined, all types of features can be relevant and it depends on the definition type which setting combination gives the best results. In all cases, the precision obtained with the pattern-based approach is improved. In particular, the results for the *punctuation* and *pronoun* definitions improve

considerably.

1.7 OUTLINE OF THE THESIS

In Chapter 2, an overview is presented of different methodologies that have been used in the past to classify definitions. The methodology that has been adopted for the classification of the LT4eL definitions is pattern-based, and takes the lexico-syntactic patterns of the definitions as its starting point. To assess how the definition types distinguished are actually used, the definitions from the LT4eL corpus have been classified on the basis of this methodology. Chapter 3 presents the pattern-based approach and describes how the lexico-syntactic patterns from Chapter 2 can be formalized using regular expressions. The results of the pattern-based approach are the input for the machine learning approach that is discussed in Chapter 4. In Chapter 4, the focus is on the selection of appropriate features for the classification of definitions. The results achieved with the different classifiers are presented in Chapter 5. The last chapter outlines the conclusions that can be drawn on the basis of the work carried out and discusses some directions that could be explored in future research on definition extraction.

'What I was going to say,' said the Dodo in an offended tone, 'was, that the best thing to get us dry would be a Caucus-race.'
'What is a Caucus-race?' said Alice; not that she wanted much to know, but the Dodo had paused as if it thought that somebody ought to speak, and no one else seemed inclined to say anything.
'Why,' said the Dodo, 'the best way to explain it is to do it.'

Lewis Carroll in *Alice in Wonderland* (1832 - 1898)

2 Definitions

2.1 INTRODUCTION

Research on definitions has a long history. The philosophers Aristotle and Plato were the first to discuss the philosophical question of what is a definition. Later research on this topic mainly focused on the different methods that can be used to define a term. These can be very diverse and range from using gestures or actions to show what something means to describing a word by using examples.

A possible definition method is illustrated in the example of the Dodo at the beginning of this chapter. The Dodo explains the meaning of a 'Caucus-race' by demonstrating what it is. Whereas in this case no language is involved, most definition methods use words to explain a term. Section 2.2 shows in which ways definitions have been classified in the course of history. The most common classification methodology has been the use of semantic properties to determine the class of a definition. The overview ends with a specification of the methodology that has been proposed to categorize the corpus definitions – the pattern-based classification methodology.

Section 2.3 describes how this methodology has been used to divide the definition patterns into four groups. The patterns of the different definition types have been examined to detect similarities and differences. This investigation constitutes the basis for the pattern-based extraction approach that will be presented in Chapter 3.

2.2 CLASSIFICATION OF DEFINITIONS

The history of definitions goes back to the 4th century BC, the time of Plato and Aristotle. From that time, the notion of definition has evolved during the centuries. For the purpose of definition extraction, it is not necessary to know the details of the philosophical debates on the question of what a definition is. Instead of discussing this, the focus is on the different definition types that have been distinguished in history from Plato to the 21st century, since for my research, which focuses on definition extraction, this is the most important aspect.

2.2.1 REAL AND NOMINAL DEFINITIONS

When Plato introduced the notion of definition in the 4th century BC, he only thought of definitions of things and not of words or symbols. His vision on definitions is illustrated in 'The Allegory of the Cave', in which he describes a fictional conversation between Socrates (Plato's teacher) and Glaucon (Plato's brother) (Plato, 514a-541b)[1]. The allegory describes a group of prisoners who have lived chained in a cave all of their lives, facing an empty wall. Things that pass by are projected as shadows on this wall and are watched by the prisoners. The prisoners begin to ascribe forms to these shadows, but do not get closer to seeing reality than the limited reflection shown by the shadows. Related to definitions, Plato states that people can only get restricted insights in the reality. He takes the concept 'Good' as an example of a concept for which it is very difficult to provide a definition. The allegory states that it is not enough to give examples to understand what 'Good' or 'Goodness' is and that there is a difference between 'Goodness' itself and its many appearances. The appearances are only shadows of 'Good', and do not refer to the concept 'Good' itself. One of the conclusions from the allegory is that it is not possible to see Goodness at all. What we see, are many good things, but not Goodness itself. From the problem of defining 'Goodness', Plato became interested in the general problem of how one can find the absolute characteristic shared by many examples that exactly defines a term. A definition is the end of the process of

[1] To refer to Plato's works, the Stephanus Pagination code has been used instead of the year

getting to know the most real things there are. These things, which he called 'Forms' or 'Ideas', constitute the basis of his 'Theory of Forms'.

Aristotle, a student of Plato, had a different emphasis than Plato when looking at definitions. According to him, the main point is that a definition is "a phrase which signifies the what-is-to-be" (Aristotle, 100a)[2,3]. In one of the six parts of the Organon, Aristotle provides a more detailed description of what he considers to be definitions (Aristotle, 71a)[4]:

> Since a definition is said to be an account of what something is, it is clear that one type will be an account of what its name, or some other name-like account, means – e.g. what triangle means. [...] One definition of definition is the one just stated. Another definition is an account which shows why something exists. Hence the former type means something but does not prove it, whereas the latter will clearly be like a demonstration of what something is, differing in arrangement from a demonstration. For there is a difference between saying why it thunders and what thunder is. In the one case you will say: Because the fire is extinguished in the clouds. But: What is thunder? – A noise of fire being extinguished in the clouds. Hence the same account is given in different ways: in one way it is a continuous demonstration, in the other a definition. Again, a definition of thunder is noise in the clouds; and this is a conclusion of the demonstration of what it is. One type of definition, then, is an indemonstrable account of what something is; another is a deduction of what something is, differing in aspect from a demonstration; a third is a conclusion of the demonstration of what something is.

The three different types of definitions described in the fragment are shown in Table 2.1. The characteristic of the first type is that no

[2]To refer to Aristotle's works, the Bekker Number codes have been used instead of the year

[3]Aristotle. Topics (100a). Book I, chapter 5, page 4

[4]Aristotle. Posterior Analytics (71a). Book II, chapter 10, page 58

type	example
indemonstrable account	A triangle is a polygon with three corners
deduction based on demonstration	Thunder is a noise of fire being extinguished in the clouds
conclusion of demonstration	Thunder is noise in the clouds

Table 2.1 *Examples of the definition types distinguished by Aristotle*

existence claim is made, that is, definitions of this type only state how to use a term and do not say anything about why the term exists. The two other types differ from the first in that they state something about the world. In the deduction based on demonstration, the definitions describe the cause of existence of a thing. Cause is understood here primarily as the final cause, which is that towards which a thing tends to be. In the third definition type, only the conclusion of the demonstration is presented, without the reason why it exists. This is an abbreviated expression of the second, the only difference being that in this type, from a complex statement of causes, only the last, proximate one is stated.

The emphasis of both Plato and Aristotle was on the second and third type from Table 2.1, as they cared especially about the existence of things and the causes for this existence. The emphasis remained the same until the 17th century. In this century, Locke and Spinoza adopted Aristotle's insights and shed new light on them. Spinoza (1677) still regarded definitions as primarily about things. To use his own words, "the true definition of each thing involves nor expresses anything apart from the nature of the defined thing"[5] (Spinoza, 1677). John Locke was the first to propose a distinction between nominal and real definitions, which has been adopted by many others later on. In his 'Essay concerning Human Understanding' (Locke, 1690), he used the word 'gold' to explain the difference between the two. The real definition of 'gold' is "the constitution of the insensible parts of that body, on which those qualities and all the other properties of gold depend" whereas the nominal definition of 'gold' is "that complex Idea of the word Gold stands for [...], a Body yellow, of a certain weight, malleable, fusible

[5]B. de Spinoza. Ethics (1677). Book 1, proposition 8, page 80

and fixed".[6] Following Locke, a rough way to mark the distinction between real and nominal definitions is thus by saying, that the former states real essence, while the latter states nominal essence. Depending on the purpose and circumstances in which a word is used, there may be a preference for either one of the two definition types. For example, chemists aim at real definitions, whereas lexicographers aim at nominal definitions (Gupta, 2008).

The division into nominal and real definitions is the result of long philosophical debates. The term 'definition' nowadays generally refers to nominal definitions, since usually the aim is to find out what a word means when one uses it and how words or phrases can be embedded into a larger framework. In this thesis, the focus is on nominal definitions as well, since it is the actual way in which words are employed that matters in the creation of glossaries. Therefore, a more detailed description of (the sub types of) nominal definitions is given in the next section.

Within the main division of definitions into real and nominal definitions, different classifications into subtypes have been proposed. Robinson (1972) makes a distinction between purpose-based and method-based definitions. In the purpose-based classification, the purpose of the definition determines the sub types (e.g. lexical definitions). The method-based categories are inferred from the ways that are employed to define terms (e.g. using examples, giving relations). Other authors do not make the distinction of Robinson (1972), although the sub types they describe are comparable. There is no author that provides a complete overview of definition types. Two authors who did an attempt into this direction are Robinson (1972) and Borsodi (1967). The classification methodologies based on the purpose and on the method will be described in Section 2.2.2 and 2.2.3.

2.2.2 PURPOSE-BASED CLASSIFICATION

In the previous section, the nominal definitions have been addressed. According to Robinson (1972) they can be further classified on the basis of the purpose. More specifically, he distinguishes *word-word* and *word-*

[6] J. Locke (1690). An Essay concerning Human Understanding. Book III, VI.2

thing definitions. A *word-word* definition is a definition which states that two words refer to the same entity. An example of this type would be saying that the Dutch word 'blauw' means the same as the French word 'bleu'. In a *word-thing* definition, the meaning of a word is said to be a a certain thing, a word is then correlated to a thing instead of a word. For example, a *word-thing* definition of the Dutch word 'blauw' could be given by pointing at a blue sky and saying that the Dutch word 'blauw' means this colour. The word-thing definitions are more common than the word-word definitions, which are most often the entries found in a simple translation dictionary. Within the *word-thing* definitions, Robinson (1972) distinguishes lexical and stipulative definitions as sub types.

Lexical definition A lexical definition is a "word-thing definition in which we are explaining the actual way in which some actual word has been used by some actual persons".[7] According to Robinson, in lexical definitions, three parties are involved: first the definer who is explaining the meaning of the word (in written texts this is the author of the text), second the hearer or reader to whom the meaning is explained, and third the person or persons whose usage of the word gave the term its meaning. Robinson (1972) considers lexical definitions to be different from dictionary definitions. The difference between lexical and dictionary definitions is in his opinion that the third party – that is, the historical aspect – is often ignored in dictionary definitions, which generally provide lexical descriptions of the vocabulary of a respected class and have a more authoritative character. As an example, Robinson provides a lexical and a dictionary definition of the word 'phlogiston'. The dictionary definition is "the hypothetical principle of fire regarded formerly as a material substance"[8] whereas the lexical description is "the cause of fire", as this is what users of the word meant when they used this term in history. However, most other researchers consider lexical and dictionary definitions to be of the same type (cf. Parry and Hacker (1991)).

[7] R. Robinson. Definitions (1972). Chapter III

[8] Merriam Webster Online dictionary. 11 September 2008

Stipulative definition A stipulative definition is the "explicit and self-conscious setting up of the meaning relation between some word and some object, the act of assigning an object to a name (or a name to an object), not the act of recording an already existing assignment".[9] The fragment from *Alice in Wonderland* (Carroll, 1999) clearly illustrates what a stipulative definition is[10]:

> '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.'
>
> 'The question is,' said Alice, 'whether you can make words mean so many different things.'
>
> 'The question is,' said Humpty Dumpty, 'which is to be master – that's all.'

So a stipulative definition, is a definition that tells what one intends a word to mean. It stipulates that a word only has the meaning stipulated in this context. This kind of definition can be used when a certain word has a specific meaning in the one context which is different from its meaning in other contexts.

2.2.3 METHOD-BASED CLASSIFICATION

Definitions can be classified according to their purpose. An alternative possibility is to examine the methods that are used to reach this purpose. Robinson (1972) distinguishes seven types of nominal definitions based on the method adopted, which can be used for both lexical and stipulative definitions. Borsodi (1967), who does not make the distinction between methods and purposes, provides a list containing nineteen nominal definition types, most of which can be described in terms of the method employed. Several others have distinguished different definition classes, however, they did not intend to provide a complete

[9] R. Robinson. Definitions (1972). Chapter IV

[10] L. Carroll. Through the Looking Glass (1999). Chapter VI

overview (cf. Johnson (1921), Pepper (1945), Bridgman (1928)). Most of their types have been included in the overviews provided by Robinson (1972) and Borsodi (1967). In what follows, the seven types of definitions proposed by Borsodi (1967) will be described in more detail.

Ostensive definitions If the meanings of all words were given only by using words, their meanings would be circular. It is thus necessary that one first gets to know the meaning of some words by using other information sources than just words. To this end, the ostensive method can be used. This method relies on more than words alone and makes use of the learner's experience of examples. Johnson (1921) was the first to recognize this method and used it mainly as "imposing a name in the act of indicating, presenting or introducing the object to which the name is to apply". In other words, he used a combination of words and objects present to explain things. Robinson (1972) distinguishes four other sub-methods within this type, in all of which more things than just words are used (e.g. drawings or pointing). The ostensive method is necessary to learn the meaning of some basic words. Other words can then be explained by referring verbally to those ostensively defined words. There is not a fixed set of words that should be defined this way, the crucial point is that there are some words defined this way to get a basis.

Synonymous definitions The method of synonymy is common in dictionaries and consists of providing the learner a synonym with which he is already familiar. Apart from such standard synonymous definitions, Borsodi (1967) mentions two other types that can be considered synonymous definitions as well, namely derivative and translational definitions. In a derivative definition, a word is defined by reference to its derivation. In English, words are often have a Latin or Greek origin. The translational definition provides a translation of an unknown word by giving a known word which has (almost) the same meaning. The analogic definition can also be seen as a type of synonymous definitions. Words are in this case defined by referring to a similar object or event which is better known to clarify its meaning (Borsodi, 1967). Table 2.2 shows an example of the four types of synonymous definitions.

sub type	example
synonymous	A foe is an enemy
derivative	definiendum (Latin: that which is to be defined)
translational	'Vijand' is the Dutch word for enemy
analogic	Hyves is something like Myspace

Table 2.2 *Examples of synonymous definition sub types*

Analytical definitions An analytical definition provides an analysis of a word or phrase to clarify its meaning. Thus, apart from letting us know what it means, the definition also provides an analysis of it. The genus-differentia definition proposed by Aristotle is an analytical definition. It first describes a word by situating it into a broader category (the genus) and then differentiates it from other words in that broad category by mentioning the distinguishing properties of it (differentiae). The analytical method is often used in dictionaries. Compared to the synonymous definitions, analytical definitions are more elaborate. However, an obvious advantage is that they provide more information than a synonymous definition.

Borsodi (1967) distinguishes five definition types that all fit into this category, namely classificatory, operational, anatomic, qualitative and quantitative definitions. A classificatory definition resembles the genus-differentia definition, but only states the class to which a referent belongs. In an operational definition, a word is defined by describing what it does or can do and for what purpose it may be used. An anatomic definition enumerates a sufficient number of the parts or organs of the word to make its whole nature clear. The qualitative definition states the qualities, aspects, characteristics, or properties of the word and the quantitative definition describes its size, weight, length, area, or any other dimension using numerical mathematical symbols. In Table 2.3, an example for each of the analytical definition types is shown.

Relational definitions The relational method introduced by Johnson (1921) explains the meaning of a term by taking into account the relations among entities. Robinson (1972) uses the term *synthetic* definition

sub type	example
genus-differentia	A footnote is a a note of text placed at the bottom of a page in a book or document that can provide an author's comments on the main text or citations of a reference work in support of the text, or both.
classificatory	Python is a programming language
operational	Gnuplot is a program for drawing graphs
anatomic	A table consists of rows and columns
qualitative	Coordinated movement is the simultaneous control along multiple degrees of freedom, which results in an efficient trajectory
quantitative	A mountain is a peak that rises over 2,000 feet (609.6 m)

Table 2.3 *Examples of analytical definition sub types*

for this method. Theoretically, this type of definition is always possible, as everything stands in a relation to other things. An advantage of this method is that it enables us to reach greater agreement and precision in stipulative definitions. The main disadvantage of the relational method is, that it does not give the meaning of the word unless either the word means something logically determined by the relation given or the learner is otherwise acquainted with the thing meant.

The relational definitions can be divided into antonymic and meronymic definitions (Borsodi, 1967). In an antonymic definition, a word is defined by enumerating contrasting words referring to referents of an opposite nature. Meronymic definitions explain a word by situating it between two other terms which refer to anything in between the synonym and antonym of the word to be defined. Table 2.4 shows an example antonymic and meronymic definition.

sub type	example
antonymic	Bad is the opposite of good
meronymic	The present is the moment in time between past and future

Table 2.4 *Examples of relational definition sub types*

Exemplifying definitions In this method, examples are used to illustrate the meaning of a word or term. An exemplifying definition of the word 'bird' is shown in Table 2.5. Although this can be a very effective method, it has at least two disadvantages. First, according to Lewis (1929) it is not possible to give a collection of cases to determine the exact meaning of a term. A second related objection also mentioned by him is the fact that it is not possible to be sure that the common element extracted is the right one. As a consequence, what is achieved by the exemplificatory method is never better than an hypothesis about the meaning of a word. It is not possible to use this method for defining all terms, e.g. mathematical terms can not be defined with only examples. Robinson (1972) uses the term denotative definitions for this type whereas Borsodi (1967) does not distinguish this type. Exemplifying definitions can be used when a list of examples provides more relevant information than can be conveyed with other types of definitions, or when listing the members of a set tells enough about the nature of that set. The latter is the case in the exemplifying definition of the word 'bird'.

type	example
exemplifying	Bird means such things as swans, robins, geese, hens and larks, and not such things as bats, butterflies or aeroplanes

Table 2.5 *Example of an exemplifying definition*

Contextual definitions A contextual definition puts the word being defined in a context to explain its meaning, as shown in the example in Table 2.6. This method differs from all others in two ways. First, because it *uses* the word being defined instead of *mentioning* it, whereas the other methods mention it and do not use it and second, there is no distinction between definiendum and definiens, as there is no phrase equivalent to the term provided. Different names have been proposed for this type. Gergonne (1818) introduced it under the name *implicative* definition, since it provides a sentence that implies what something means whereas Borsodi (1967) calls it an *illustrative* definition, since it

gives an illustration of a sentence or phrase in which the word being defined is used.

type	example
contextual	A square has two diagonals, and each of them divides the square into two right-angled isosceles triangles

Table 2.6 *Example of a contextual definition*

Reference definitions A reference definition is a definition in which the author refers to another source of information (e.g. a document or a person). Borsodi (1967) distinguishes three types of reference definitions, namely descriptive, historical and quotational definitions. The descriptive and historical definitions are intended for defining proper names only. In the descriptive definition, a person describes a proper name on the basis of his personal experience. On the other hand, the historical definition uses a historical document to define a name. The third type differs from the others in that it provides a definition of a word instead of a proper name. It does so by quoting one or more phrases in which the word being defined is used by some authority and its meaning is made clear by the context. The example on eLearning provided in Table 2.7 is such a quotational definition.

type	example
quotational	According to the EU, eLearning involves the use "of new multimedia technologies and the Internet to improve quality of learning by facilitating access to resources and services as well as remote exchanges and collaboration" (www.elearningeuropa.info)

Table 2.7 *Example of a quotational reference definition*

Rule-giving definitions In all previous methods, definitions are used to define a particular (e.g. Utrecht) or a general thing (e.g. goodness). However, there are also words that do not refer to one thing, but have a different meaning depending on the circumstances in which it is used.

These are words like 'him', 'there', 'soon', and 'yesterday'. They are defined by rule-giving definitions, since it is possible to give general rules that describe what they mean and when they are appropriate. This definition type is quite rare. Generally, such words are learnt by observing others using them (Robinson, 1972). Dictionaries provide definitions of such words by describing the way in which they are used (that is, the rule for using them) and providing some examples. Table 2.8 shows a rule-giving definition of the word 'he' from the Longman Dictionary of English Language and Culture.

type	example
rule-giving	**he** *pron* (used as the subject of a sentence) **1** that male person or animal already mentioned: "Where's John?" "He's gone to the cinema" | "Be careful of that dog - he sometimes bites"

Table 2.8 *Example of a rule-giving definition*

2.2.4 PATTERN-BASED CLASSIFICATION

The method-based classification methodology is a semantic framework that uses the way in which information is conveyed to distinguish definitions from each other. The methodology provides valuable insights on the various ways that can be employed to explain the meaning of terms. However, the fact that semantic properties are difficult to formalize constitutes a problem for the glossary creation application in which the aim is to develop an approach that can extract definitions automatically from texts. Since extraction on the basis of the method-based methodology will be very complicated, we decided to investigate whether an alternative approach could be used.

We noticed that a restricted set of lexico-syntactic patterns can be used to cover the majority of definitions. For computers, such patterns are easier to formalize and deal with than semantic information. Therefore, (Westerhout and Monachesi, 2007a) proposed another methodology for the categorization of the Dutch definitions that takes the lexico-syntactic patterns of definitions as a starting point for the classification

instead of semantic properties.[11]

The pattern-based classification methodology relies on the assumption that all definitions consist of (at least) three elements, namely a definiendum, a definiens and a connector (Walter and Pinkal, 2006). Figure 2.1 shows an example definition in which these three constituents are present. The definiendum, often the subject, is the element that is defined (Latin: *that which is to be defined*). This is the term 'definition' in the example. The definiens provides the meaning of this definiendum (Latin: *that which is doing the defining*). In the example sentence, the definiens is 'a statement of the meaning of a word'. The definiendum and the definiens are connected via the connector, which can be either a verbal phrase or a punctuation character. This connector indicates what the relation is between definiendum and definiens. The connector *is* has been used in the example.

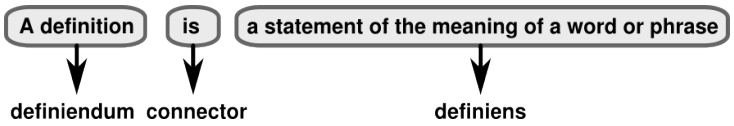

Figure 2.1 *Elements of a definition*

The classification methodology based on the patterns of definitions has been proposed within the scope of the LT4eL project (Westerhout and Monachesi, 2007a). It divides the definitions in six categories, which appeared to be language independent. An example for each of the definition types is provided in Table 2.9. All examples have been taken from the Dutch LT4eL corpus, since the focus in this thesis is on the extraction of Dutch definitions.

[11] Another typology that pays attention to lexico-syntactic patterns of definitions to some extent is the genus-differentia typology from Sierra et al. (2008). Since their work has been done in parallel with my work, I only came across this work after the experiments had been completed.

type	example
is	Gnuplot is een programma om grafieken te maken. *'Gnuplot is a program for drawing graphs.'*
verb	E-learning omvat hulpmiddelen en toepassingen die via het internet beschikbaar zijn en creatieve mogelijkheden bieden om de leerervaring te verbeteren. *'eLearning comprises resources and applications that are available via the Internet and provide creative possibilities to improve the learning experience.'*
punctuation	Passen: plastic kaarten voorzien van een magnetische strip, die door een gleuf gehaald worden, waardoor de gebruiker zich kan identificeren en toegang krijgt tot bepaalde faciliteiten. *'Passes: plastic cards equipped with a magnetic strip, that can be swiped through a card reader, by means of which the identity of the user can be verified and the user gets access to certain facilities. '*
pronoun & connector	Dedicated readers. Dit zijn speciale apparaten, ontwikkeld met het exclusieve doel e-boeken te kunnen lezen. *'Dedicated readers. These are special devices, developed with the exclusive goal to make it possible to read e-books.'*
pronoun as connector	In het dagelijkse leven schrijven we getallen in het tientallig of decimale stelsel, d.w.z. dat we 10 cijfers gebruiken (0 t/m 9). *'In everyday life we write numbers using the decimal system, i.e. we use 10 digits (0 through 9).'*
layout	RABE Een samenwerkingsverband van een aantal Duitse bibliotheken, die gezamenlijk een Internet inlichtingen dienst bieden, gevestigd bij de gemeenschappelijke catalogus, HBZ, in Keulen. *'RABE* *A cooperation of a number of German libraries, that together provide an Internet information service, residing at the common catalog, HBZ, in Cologne.'*
other	Bij superscript wordt de tekst iets hoger afgedrukt dan normaal en bij subscript iets lager. *'In superscript text is printed slightly higher than normal and in subscript slightly lower.'*

Table 2.9 *Examples of pattern-based definitions*

The first three types are purely based on the connector phrase they contain. These are the *is* definitions, the *verb* definitions, and the *punctuation* definitions. In the *is* definitions, the connector is a form of the copula verb 'to be'. In the *verb* definitions, a verb or verbal phrase is used to connect definiendum and definiens and to indicate how these two parts are related to each other. The *punctuation* definitions contain a punctuation character that connects definiens and definiendum. This punctuation character is usually a colon or bracket, in some cases it is a comma or a dash.

The fourth type are the pronoun definitions, which can be divided into two distinct groups. For both types, an example is presented in Table 2.9. In the first, the pronoun 'dit' (*these*) refers back to a previously used definiendum or definiens. In this case, the definition contains both a pronoun and a connector. The other *pronoun* definition includes only a pronoun phrase, such as the abbreviation 'd.w.z.' (*i.e.*), as the connector. In this case, the pronoun phrase itself links the definiendum and the definiens to each other.

The fifth type is different from the others, since here the layout of the sentence is the only indication that it might be a definition. In the example in Table 2.9, the definiendum ('RABE') is the heading of the section and the definition is the first sentence of the section. Another layout feature typical for layout definitions are tables where the definiendum is contained in the leftmost cell of a row and the definiens in the cell to the right of the definiendum. The last type are the unclassifiable definitions, these are definitions in which either an uncommon connector verb is used or another pattern that is not common for definitions. In the method-based classification methodology, these definitions could be classified as exemplifying definitions.

2.3 CLASSIFYING THE CORPUS DEFINITIONS

In this section, the definitions from the Dutch part of the LT4eL corpus are investigated and classified on the basis of the pattern-based classification methodology from Westerhout and Monachesi (2007a). The definitions have been closely examined to discover which characteristics are shared among the definition patterns. This analysis constitutes

the basis for the development of the pattern-based extraction approach that will be discussed in Chapter 3.

type	**definitions**	%
is	186	29.0
verb	201	31.4
punctuation	112	17.5
pronoun	104	16.2
layout	7	1.1
other	31	4.8
TOTAL	641	100.0

Table 2.10 *Classification of the Dutch LT4eL definitions on the basis of the pattern-based methodology*

Table 2.10 provides the frequencies of the different types in the LT4eL corpus. As can be seen, the majority of definitions can be classified in one of the first four definition categories. There are 75 definitions that consist of more than one sentence. The majority of the multi-sentence definitions (89.3%) belongs to the *pronoun* category. In those pronoun multi-sentence definitions, one of the sentences only contains the definiens (52 times) or definiendum (16 times) whereas the other sentence contains the remaining definition elements, that is, the connector and either the definiendum or the definiens. The remaining multi-sentence definitions are actually combinations of two definitions, since both sentences conform to a definition pattern.

The *layout* and *other* definitions differ from the other types, since they do not contain a lexico-syntactic pattern indicating that it is a definition. In addition, these types are both very uncommon in the corpus compared to the other definition types. For these reasons, they have not been addressed in my experiments and are not included in the results.In this thesis, the focus is on the *is*, *verb*, *punctuation*, and *pronoun* definitions, which all contain linguistic characteristics that are typical for definitions. The patterns of definiendum and definiens are more or less comparable, but the types are different with respect to their connectors phrases. In the *pronoun* definitions, a pronominal pattern functions as an additional indicator.

2.3.1 *Is* DEFINITIONS

An *is* definition is a definition in which the verb 'to be' is used as the connector. The corpus contains 186 definitions of this type of which 185 consists of one sentence.

(2) a. Gnuplot is een programma om grafieken te maken .
Gnuplot is a program to graphs to draw .
'Gnuplot is a program for drawing graphs.'

b. Liegen is een manier om mensen te misleiden .
Lying is a way to people to cheat .
'Lying is a way to cheat people.'

Example 2a shows a typical *is* definition from the corpus. The sentence is an example of a broad definition, since the definiendum can be replaced with other graph drawing programs, such as Graphviz, without turning the sentence into an incorrect statement. The definiens thus does not provide a unique description for exactly this term. However, in the application of glossary creation it is a relevant definition since the definiens gives quite valuable information about the definiendum for people who do not know what Gnuplot is. Many of the LT4eL *is* definitions are broad definitions.

An important problem with the *is* definitions is that there are many non-definitions that use the verb *is* as a main verb as well. The lexico-syntactic patterns of such sentences are often similar to the pattern of *is* definitions. Example 2b illustrates this problem and shows a sentence that has the same pattern as example 2a. However, only example 2a is a definition, although example 2b provides useful information about the meaning of lying. It is questionable whether it will be possible to distinguish example 2a from 2b on the basis of its lexico-syntactic structure. Human evaluators use other information – such as world knowledge – in combination with additional linguistic information to make their judgements.

The majority of *is* definitions (89.3%) have a pattern that is similar to the pattern of example 2a. It starts with a noun phrase as the definiendum (*Gnuplot*). The present tense of the verb 'to be' is then used as the connector and the definiens (*a program for drawing graphs*) constitutes the rest of the sentence. More generally, the definiendum of *is*

definitions is usually a noun phrase (either starting with an article or not) while the definiens in most cases begins with a noun phrase.

(3) a. Een veelgebruikt programma om presentaties te maken
A common program to presentations to make
is Powerpoint .
is Powerpoint .
'*A common program to make presentations is Powerpoint.*'

b. LaTeXis noch een DeskTop Publishing pakket noch een
LaTeXis neither a DeskTop Publishing package nor a
tekstverwerker , maar een zet-systeem .
text editor , but a typesetting system .
'*LaTeXis neither a DeskTop Publishing package nor a text editor, but a typesetting system.*'

In the remaining 10.7%, either the definition and definiens appear in the reverse order or the sentence construction is more complex. Example 3 provides an example of both types. The complex patterns are quite diverse and uncommon. Generally, the corpus contains only one or two similar examples for the same pattern.

2.3.2 *Verb* DEFINITIONS

Verb definitions are definitions in which a verb or verbal phrase (except a form of *to be*) is used as the connector between definiendum and definiens. The corpus contains 201 definitions of this type. Example 4 shows three *verb* definitions from the LT4eL corpus in which the connector phrase is used differently.

(4) a. Cryptografie wordt gebruikt om de inhoud van
Cryptography is used to the content of
elektronische documenten te verbergen of te verifiëren
electronic documents to hide or to verify
en om bestanden te beschermen tegen ongeoorloofde
and to files to protect against illegal
toegang , wijzigingen en diefstal .
access , changes and theft .
'*Cryptography is used to hide or verify the content of electronic documents and to protect files against illegal access, changes and*

theft.'

b. Een nauwkeurig gedefinieerde serie woorden ter
A precise defined sequence words for
beschrijving van een bepaald levensgebied noemt men een
description of a certain area call they an
ontologie .
ontology .
'*A precise defined sequence of words to describe a certain area is called an ontology.*'

c. Met alinea-afstand wordt de afstand tussen twee of
With paragraph space is the distance between two or
meer alinea's bedoeld .
more paragraphs meant .
'*With paragraph spacing the distance between two or more paragraphs is meant.*'

Example 4a illustrates the use of the connector phrase 'wordt gebruikt om' (*is used to*). The sentence begins with a definiendum, which is followed by the connector phrase and the definiens. Most *verb* definitions have this pattern, although there are also patterns in which the order of definiendum and definiens is different. The first alternative is the reverse order of the standard pattern, that is, definiens – verb (phrase) – definiendum (example 4b). In this case, the connector verb ('noemen' (*to call*)) appears after the definiens and before the definiendum. In the other option, illustrated in example 4c, the definiendum is embedded in a prepositional phrase at the beginning of the sentence.

2.3.2.1 Verbal connector phrases

Based on the corpus, a list of 35 different verbs that are exploited in connector phrases has been compiled. A distinction can be made between connectors consisting of only a verb (e.g. 'betekenen' (*to mean* or 'noemen' *to name*) and connector phrases in which a verb is combined with one or more other words (e.g. 'bestaan uit' (*to consist of*) or 'fungeren als' (*to function as*)). Some of the connector verbs are used in more than one connector phrase. For example, the verb 'gebruiken' (*to*

use) is used in definitions in two ways, namely as 'gebruiken om' (*to use to*) and as 'gebruiken voor' (*to use for*). Since connectors can consist of more than one word, the term 'verbal connector phrases' or simply 'verbal phrases' is used to denote connector phrases in which a verb is used. In total, there are 50 verbal phrases.

Twelve *verb* definitions (6.0%) do not contain a verbal connector phrase indicating that the sentence is a definition. More details on these patterns are provided in Section 2.3.2.2. In the remaining 189 *verb* definitions, one of the 50 definition verbal phrases is used. This means that each verbal phrase is used on average 3.8 times. However, the distribution of the phrases is not equally divided over the 37 possible phrases, but resembles a Zipf curve: there are some frequent patterns and there is a relatively large tail of patterns occurring only once or twice in the corpus.

The most common individual connector verb is 'gebruiken', which accounts for 16.9 % of the *verb* definitions. Other frequently used verbs are 'bestaan' (*to consist*), 'betekenen' (*to mean*) and 'noemen' (*to call*). There are 12 verbs that have been used only once. These are verbs like 'beschrijven' (*to describe*), 'verklaren' (*to explain*) and 'behelzen' (*to include*). If one takes the verbal connector phrases into consideration instead, the most common frequently used phrase is 'bestaan uit' (*to consist of*), which is used in 10% of the definition sentences. This phrase is followed by the phrases 'gebruiken om' (*to use to*), 'noemen' (*to call*) and 'gebruiken voor' (*to use for*).

There are 23 phrases that are used only once. In addition to the before-mentioned verbs, these include phrases like 'mogelijk maken om' (*to allow to*), and 'mogelijk maken door' (*to make possible by*). Although these uncommon phrases are less clear connector phrases than the more frequent ones, they are nevertheless relevant in a glossary creation context, since the definiens in these sentences provides very useful information about the definiendum. As has been said earlier, within the application of definition extraction in the domain of glossary creation it is crucial to retrieve as many definitions as possible. Figure 2.2 provides an overview of the occurrences of the different verbal connector phrases in the corpus.

Whereas in most connector phrases the simple present tense of an

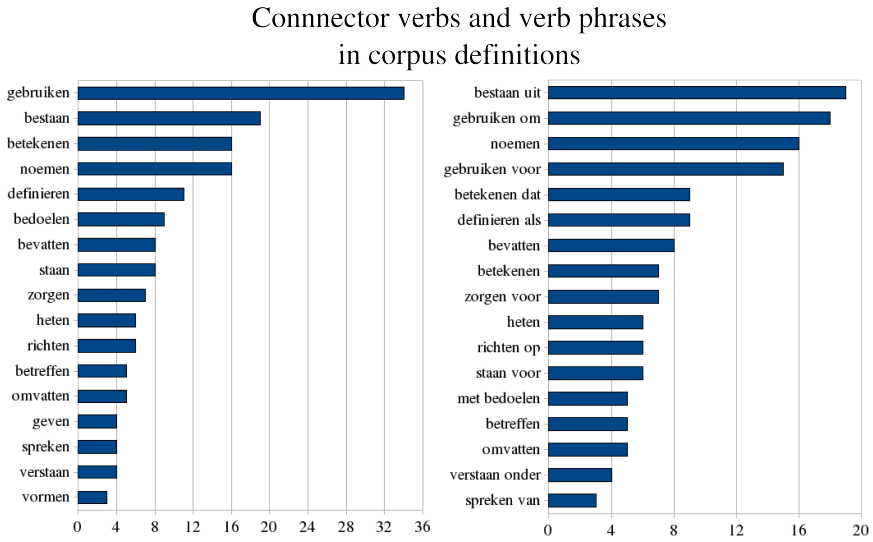

Figure 2.2 *Verbs and verbal connector phrases occurring three times or more in the corpus*

independent verb is used as the main verb, there are also *verb* definitions in which an auxiliary verb is embedded within the connector phrase. This happened in 35.4% of the definitions. In 25.9% of the definitions, the passive auxiliary verb 'worden' (*to be*) has been used. The auxiliary verbs 'zijn' (*to be*, 6.9%), 'kunnen' (*can*, 4.2%), and 'zullen' (*will*, 0.5%) are used considerably less. Two example definitions in which the verbal connector phrase contains the verb 'worden' have been shown in example 4a and 4c.

2.3.2.2 Problematic patterns

A number of patterns that have been tagged as *verb* definitions differ from the *verb* definitions discussed previously, since they do not contain a verbal phrase which indicates that the sentence is a definition. In the corpus, there are twelve sentences for which this is the case. Example 5a illustrates the problem.

(5) a. Een vaste spatie voorkomt dat een regel tussen twee
A fixed space prevents that a line between two
woorden wordt afgebroken .
words is broken down .

'*A non-breaking space prevents that a line is broken down between two words.*'

b. Een vaste spatie is een spatie die voorkomt dat een regel
A fixed space is a space that prevents that a line
tussen twee woorden wordt afgebroken .
between two words is broken down .

'*A non-breaking space is a space that prevents that a line is broken down between two words.*'

Example 5a contains no lexical or syntactic evidence that indicates that it is a definition. It is hard to extract such definitions using a pattern-based approach. Example 5b illustrates that a slightly adapted version of the sentence would be an *is* definition. The change generally involves adding a small phrase after the definiendum – 'is een N die/dat' (*is a(n) N that*) – and then adapting the rest of the sentence to make it grammatically correct. Such definitions are not considered in my extraction approach.

2.3.3 *Punctuation* DEFINITIONS

Within the *punctuation* patterns, four types of characters can be used as the connector. By far the most frequent type (65.2%) is the 'colon definition', in which a colon is used to connect definiendum and definiens (Example 6a). In the second type (Example 6b), the 'bracket definitions', brackets are used around either the definiendum or the definiens. The bracket definitions account for 29.5% of the *punctuation* definitions. Apart from these two types, there are two other less common *punctuation* definitions: in the 'comma definitions' (Example 6c), the connector is a comma (4.5%) and in the 'dash definitions' (Example 6d), a dash is used to this end (0.9%).

(6) a. 500 Internal server error : een fout in de server
500 Internal server error : an error in the server
waardoor hij niet kan reageren
as a result of which he can not react
'*500 Internal server error : an error in the server which prevented it from reacting*'

b. ' antecedent ' (het woord waarnaar verwezen wordt)
' antecedent ' (the word to which referred is)
''*antecedent' (the word that is referred to)*'

c. een IRC-client , een programma dat ontworpen is om
an IRC client , a program that designed is to
verbinding te maken met een of meerdere IRC-netwerken
connection to make with one or more IRC networks
.
.
'*an IRC client, a program that is designed to connect to one or more IRC networks.*'

d. standaards - gecodificeerde regels en richtlijnen voor de
standards - encoded rules and guidelines for the
creatie , omschrijving en het beheer van digitale
creation , description and the management of digital
bronnen
sources
'*standards - encoded rules and guidelines for the creation , description and management of digital sources*'

2.3.3.1 Colon definitions

Renkema (2002) presents in his book 'Schrijfwijzer' (*Writing guide*) four situations in which a colon should or could be used in Dutch:

1. before an enumeration
2. to announce a description, quotation, explanation or conclusion
3. after a sentence starter (e.g. 'Samengevat' (*Summarized*), 'Dat wil zeggen' (*That is*), 'Met andere woorden' (*in other words*)
4. in combinations with numbers (e.g. '10 : 2 = 5' or 'Genesis 11:9').

The way in which the colon is used in the *punctuation* definitions fits best into the second category, that is, to announce a description, quotation, explanation or conclusion. However, since the colon for a definition is not the only way of using this character, the extraction of colon definitions will be a difficult task; it is not possible to simply extract all sentences containing a colon.

The corpus contains 73 colon definitions. Except for one definition, they all start with the definiendum after which the colon and definiens follow. Compared to the *is* and *verb* definitions, the definiendum begins less often at the first position of the sentence (82.2% of the definitions). In almost half of the cases, a noun phrase is used directly after the colon. The second most frequent phrase in this position is an adverb (16.4 %). Other common phrases are pronouns (13.7%) (most times the demonstrative pronoun 'dit' (*this*)) and verbs (12.3%). The other syntactic categories used at the first position of the definiens all occur less than 5 times (e.g. conjunctions or preposition phrases).

2.3.3.2 Bracket definitions

Brackets are punctuation marks used in pairs to set apart or interject text within other text. It is often possible to leave the part between brackets out of the sentence, because in some cases it only adds extra information which can be left out without problems. There are four types of brackets:

- round brackets or parentheses: ( )
- square brackets or box brackets: [ ]
- curly brackets or braces: { }
- angle brackets, diamond brackets or chevrons: $< >$

In the corpus definitions, only the round and square brackets are observed. The majority of bracket definitions contains parentheses while the square brackets are employed as the connector in the remaining definitions. For the parentheses, Renkema (2002) mentions a variety of possible ways in which they can be used:

1. to add or explain something,
2. in references,
3. meaning 'or',
4. to introduce an abbreviation,
5. in bibliographies, and
6. in telephone numbers.

The first use case applies to the use of brackets in definitions, that is, to add or explain a word or phrase. In most definitions from the corpus, the definiendum is mentioned before the opening bracket and the definiens is contained between the brackets. However, the reverse order can be used as well, which happens in 9.1% of the bracket definitions. In these cases, the definiens is mentioned before the brackets and the definiendum follows between the brackets.

Only 57.6% of the bracket definitions start at the beginning of the sentence. The definiendum is in all cases a noun phrase. There is a large variety of patterns within the brackets. In some cases, it is a complete sentence whereas other times it is only a noun phrase. When looking at the syntactic category of the first phrase within the brackets, the noun phrase is the most common category (57.6%) while the preposition phrase is the second most frequent category (12.1%). All other types are used in less than 10% of the definitions.

According to Renkema (2002), square brackets should only be used to add comments to a quoted or translated text. However, the corpus definitions reveal that theory can be different from practice and that many authors do not adhere to the writing guidelines; in the corpus there are a number of definitions in which square brackets are used. Since the structural characteristics of the patterns are comparable – regardless of the brackets used – both types are referred to as 'bracket definitions'.

2.3.3.3 Comma and dash definitions

In addition to the two main categories, there are two very types of *punctuation* definitions that are rarely used. In the corpus, the 'comma defi-

nition' (example 6c) has been used five times and the 'dash definition' (example 6d) occurred only once. It is therefore almost impossible to generalize over these patterns. Besides, the comma is very common in non-definitions as well; it will be a complicated task to distinguish comma definitions from non-definitions containing a comma. The definiens is, in all comma and dash definitions, a noun phrase that varies in length between 2 and 15 words.

2.3.3.4 Problematic patterns

The presence of problematic patterns is mainly caused by the fact that the corpus contains a diversity of text types. The texts in the corpus have been written for different purposes, by different authors and on different subjects. For example, the structure of manuals to learn how to use a program differs from the structure of descriptive documents. As a consequence, the patterns of definitions also differ for each text. *Punctuation* patterns that should not be used according to the writing guidelines prescribed by Renkema (2002) are nevertheless present in the LT4eL corpus.

A second problem that shows up when dealing with punctuation patterns is that the order of definiendum, connector and definiens varies. Although the pattern with the order 'definiendum – connector – definiens' is by far the most frequent pattern, there are also situations in which definiendum and definiens are switched and the structure becomes 'definiens – connector – definiendum'.

2.3.4 *Pronoun* DEFINITIONS

Pronoun definitions are definitions in which a pronoun or a phrase containing a pronoun is used to refer to a definiendum or definiens. The majority of *pronoun* definitions consists of two sentences of which the first contains only the definiendum or definiens. The second sentence contains the connector and the rest of the definition (i.e. either definiens or definiendum). The focus is on the detection of the sentence containing the pronoun and the connector. The part containing only the definiendum or definiens without a connector is much more difficult to detect, since this part does not contain any evidence that suggests

that it is part of a definition. Example 7 shows two examples of *pronoun* definitions.

(7) a. Eén van de interessantste mogelijkheden van
One of the most interesting possibilities of
Powerpoint is het werken met ' aangepaste animaties ' .
Powerpoint is the working with ' adapted animations ' .
Hiermee kunt u grafieken , figuren , schema's en
With this can you diagrams , figures , schemes and
dergelijke stap voor stap opbouwen .
the like step by step build .
'*One of the most interesting possibilities of Powerpoint is working with ' custom animations ' . With this you can build diagrams , figures , schemes and the like step by step.*'

b. U kunt in het document helptekst opnemen die bedoeld
You can in the document help text include that meant
is voor uw chef of voor uzelf . Word noemt deze tekst
is for your boss or for you . Word calls this text
aantekeningen .
notes .
'*You can include help text in your document that is meant for your boss or for yourself . Word calls such texts notes.*'

In example 7a, the definiendum is a noun phrase that is used at the end of the first sentence ('custom animations') while the connector and pronoun are in the second sentence. In principle, the definiendum can be any noun phrase in the first sentence, regardless of its position. Since almost every sentence in a text contains noun phrases, it makes no sense to take this as a requirement. It would thus be necessary to use a more sophisticated approach to identify co-reference relations in order to make detection of the definiendum possible. An indicator might be the apostrophes that are used around the definiendum in example 7a. Example 7b illustrates a different pronoun pattern. In this case, the word 'notes' is defined in the first sentence while the definiendum is introduced in the second sentence, accompanied by a pronoun ('with this') and a connector ('calls'). Again, the detection of the second sentence is easier because of the presence of the connector and the pronoun.

To detect the phrase to which the pronoun refers, it would be necessary to include co-reference detection. The problem of detecting co-reference relations is a relevant one and it has various application areas, such as summarization, question answering and information extraction. Within the COREA project, research has been done on co-reference resolution for Dutch and a system has been developed for assigning co-reference relations automatically (Hendrickx et al., 2008). However, the focus of this project was mainly on detecting co-reference relations between noun phrases (pronouns, common nouns, names) only. In addition, the use of 'het' (*it*) as a co-referent has been investigated, but the COREA system does not address this type (Hoste et al., 2007).

Many of the co-reference relations present in the pronoun definitions from the LT4eL corpus differ from the relation types addressed within COREA. For those pronoun definitions in which a form of 'dit' (*this*), 'dat' (*that*) or 'deze' (*these*) has been used to refer back to a simple noun phrase, it was investigated to which extent the COREA system was able to detect the co-reference relations[12]. Table 2.11 shows the results obtained with COREA on these definitions. The first column in this table describes the pattern of the co-reference relation. The pattern "np ... dit | dat | deze" refers to sentences in which a sole pronoun refers back to an NP. In the other pattern, the pronoun is followed by a noun (e.g. 'Dit programma' (*this program*). The second column indicates the number of times the pattern occurs in the set of definitions. Column 3, 4 and 5 give information on which parts are detected by the COREA system as being possible candidates involved in co-reference relations. From the results, it can be seen that although the individual parts are detected successfully for most sentences, it is not an easy task to find the coreference relation as well (column 6). Only 2 of the simple patterns have been successfully detected by the system. In both cases in which the reference relation was found, the pronoun was followed by a noun.

The other pronoun patterns include, among others, sentences in which 'het' (*it*) refers back to a noun phrase, sentences in which the pronoun refers to a larger phrase (e.g. a more complex noun phrase)

[12]The online demo of COREA has been used to perform the test: http://www.cnts.ua.ac.be/cgi-bin/iris/corea2demo.client.pl

		Parts detected			Relation found	
pattern	#	both	only NP	only Pron	yes	no
np(definiendum) ... this	25	18	1	6	0	25
np(definiendum) ... that NP	3	3	0	0	2	1
Total	28	21	1	6	2	26

Table 2.11 *Accuracy of COREA on LT4eL pronoun definitions*

or patterns in which the pronoun is embedded in a form of the phrase 'dat wil zeggen' (*that is*). At the moment, a system that is able to resolve these relations does not exist for Dutch.

(8) a. Dit is een operator die twee teksten aan elkaar plakt .
This is an operator that two texts together glues .
'*This is an operator which glues two texts together.*'

b. Dit initiatief bevat een overzicht van belangrijke
This initiative contains an overview of important
nationale digitaliseringsprojecten .
national digitization projects .
'*This initiative contains an overview of important national digitization projects.*'

c. De tekening is een vector-tekening , dat wil zeggen dat
The drawing is a vector drawing , that will say that
alle objecten opgeslagen worden in de vorm van
all objects saved are in the form of
hoekpunten en dergelijke .
vertices and the like
'*The drawing is a vector drawing, which means that all objects are saved in the form of vertices and the like.*'

The corpus contains 104 pronoun definitions which can be divided into three smaller groups. Example 8 provides an example for each of the three pronoun definition types. The structure of the first two types resembles the structure of *is* or *verb* definitions, except that the definiendum has been replaced by a pronoun (phrase). Within the verbal pronoun definitions, there are some sentences in which the pronoun replaces the definiens. In such situations, the previous sentence contains

the definiens. On the basis of the resemblance to the 'is' and 'verb' patterns these two types will be referred to as '*pronoun-is* definitions' and '*pronoun-verb* definitions'. In addition to these two types of patterns, there are also pronoun patterns having a completely different structure, namely the patterns in which the pronoun phrase itself functions as the connector. These definitions are called 'pronoun-connector definitions'.

2.3.4.1 Pronoun-is definitions

The corpus contains 24 pronoun-is definitions (23.1%). These are pronoun definitions in which a form of 'zijn' (*to be*) is used as the connector. The majority of pronoun-is definitions starts with the word 'dit' (*this*), either used as a demonstrative pronoun (*This is ...*) or as a demonstrative determiner (*This <Noun> is ...*). In Dutch, these are both sub types of the demonstrative pronouns. The demonstrative pronouns are referred to as 'independent' demonstrative pronouns in Dutch whereas the demonstrative determiners are called 'attributive' demonstrative pronouns. The remaining pronoun-is definitions begin with the neuter personal pronoun 'het' (*it*).

The first phrase containing the pronoun is directly followed by either the singular or plural form of the verb 'zijn' (*to be*) which is succeeded by the definiens. The patterns of the definiens are similar to the ones observed in the *is* definitions. Usually, the definiens begins with a noun phrase (83.3%), alternatively an adverbial phrase is used (12.5%) between the connector and the definiens (e.g. 'gewoon' (*just*) or 'eigenlijk' (*in fact*)). More specifically, these words indicate that the definiendum is comparable to known concepts.

2.3.4.2 Pronoun-verb definitions

Just as the frequency of *is* and *verb* definitions is comparable, the amount of pronoun-is and pronoun-verb definitions is almost the same as well; there are 26 pronoun-verb definitions (25.0%). In the pronoun-verb definitions, a larger variety of pronouns is employed than in the pronoun-is definitions. Twelve definitions contain the word 'dit' (*this*), either as a demonstrative pronoun or as a demonstrative determiner. Other pronouns that have been used are, among others, the demonstrative

pronoun 'dat' (*that*), the neuter personal pronoun 'het', and the phrase 'zo'n' (*such a*).

The verbal connector phrases employed in the pronoun-verb definitions constitute a subset of the patterns used in the *verb* definitions. Just as in the *verb* definitions, the order of the definition elements varies in the pronoun-verb definitions; the pronoun refers back to either the definiendum or the definiens. The pronoun sentences begin in both cases with a pronoun phrase followed by a verbal phrase. The sentence ends with either the definiens or the definiendum.

The verb 'noemen' (*to call*) is the most common connector in the pronoun-verb definitions (57.7%). However, since there are only 26 pronoun-verb definitions, it is very likely that new data will contain different verbal connector phrases. Including only the patterns observed in the corpus definitions might cause problems when the grammar would be applied on unseen data. A solution to this problem would be to include all verbal patterns of the *verb* definitions described in Section 2.3.2 as pronoun-verb connector patterns. The fact that the set of connector phrases used in the pronoun-verb definitions is a subset of the connector phrases used in the *verb* definitions supports this idea.

2.3.4.3 Pronoun-connector definitions

The set of pronoun-connector definitions contains 46 definitions which can be divided into two sub groups. The first group (52.2 %) consists of patterns in which a pronominal adverb starting with 'waar' (*where*), 'hier' (*here*), or 'daar' (*there*) is used as the connector, such as 'waarin' (*in which*), 'waarmee' (*with which*), 'hierin' (*in here*), 'hierop' (*upon this*), and 'daarin' (*in it*). It usually takes two words to translate these words from Dutch to English. Although literal translations for some of these terms exist – *wherein*, *wherewith whereby*, and *whereupon* – these are often seen as archaic, and as a consequence they are not common in written texts.

In the second group (45.9%), the phrase 'dat wil zeggen' (*that is (to say)*) indicates that the sentence might be a definition. The abbreviations 'd.w.z.' or 'dwz' (*i.e.* or *ie*) are alternatives of this phrase and have been used in the corpus definitions as well. In most cases, 'dat wil zeggen' functions as the connector of definiendum and definiens

within one sentence, but the phrase has also been employed at the beginning of a sentence a number of times. In these cases, the definiendum is mentioned in the preceding sentence.

2.4 CONCLUSIONS

This section concludes the chapter on the classification of definitions. An overview of a number of classification methodologies that have been proposed in the past has been presented in this chapter. The most commonly used classification scheme is the method-based one, in which semantic properties of the definitions determine the definition class. Based on the (incomplete) overviews from Borsodi (1967) and Robinson (1972), the different types of this method-based scheme have been examined. The second typology is the pattern-based methodology that has been proposed in the LT4eL project (Westerhout and Monachesi, 2007a), which was designed because it is more suitable for the automatic extraction of definitions.

The pattern-based methodology has been used to classify the Dutch LT4eL definitions into four groups. A description of the distinctive features of the patterns revealed that the *is* and *verb* definitions are more common than the *punctuation* and *pronoun* definitions. Some general observations are that in the *is*, *verb* and *punctuation* definitions the definiendum usually is a noun phrase and that the verb usually is used in third person, present tense. For the *pronoun* definitions, three types of patterns can be identified. In the first two types, the connectors are similar to the connector phrases employed in *is* and *verb* definitions. In the third type of *pronoun* definitions, the connector is a pronoun.

The next chapter describes the design and use of the pattern-based approach. The patterns of the definition types presented in this chapter constitute the basis for the regular expressions that are employed in the grammars to match definitions.

It is well to remember that grammar is common speech formulated

William Somerset Maugham (1874-1965)

3

Pattern-based definition extraction

3.1 INTRODUCTION

The investigation of the previous chapter revealed that the majority of definitions conforms to a restricted set of patterns. In these patterns, the connector phrase is of decisive importance, since this part determines whether or not a sentence can be a definition. The other definition parts that are relevant are the structure of the definiendum and the beginning of the definiens. This information can be used to develop a pattern-based method for the extraction of definitions. In such an approach, the lexico-syntactic patterns from definitions are taken as starting point.

The existing pattern-based approaches within question answering, dictionary building and ontology creation can be divided in two groups. Some approaches focus on the detection of common phrases (Tanev, 2004; Storrer and Wellinghof, 2006; Velardi et al., 2008). These phrases often correspond to the connector phrases that have been proposed for the classification of definitions in the previous chapter. Others use information on the lexico-syntactic patterns of definitions in addition to the common phrases (Muresan and Klavans, 2002; Han and Kamber, 2006). Our approach is based on a combination of connector phrases and lexico-syntactic information. The purpose is to match as many definition patterns as possible using grammars that consist of a set of regular expressions. For each of the four types of definitions, a separate grammar has been build.

In order to extract definitions from texts, it is necessary that the documents are pre-processed in a number of ways. The pre-processing steps involve the conversion of the original documents (PDF, DOC,

HTML) into a common XML format and the addition of linguistic annotation. The linguistic annotation is necessary because our extraction method relies on this informationThe investigation of the definitions described the previous chapter is the starting point for the formalization of regular expressions that match the definition patterns. The finite state transducer *Lxtransduce* has been used to identify sentences on the basis of these expressions.

The grammars have been evaluated on the corpus. The results show that it is possible to extract most definitions that are present in a text with this approach. However, they also reveal that matching a definition pattern does not automatically imply that a sentence is a definition. In other words, definitions use a restricted set of lexico-syntactic patterns, but these patterns do not uniquely identify definitions. Additional information seems to be necessary for the proper classification of sentences.

This chapter starts with an explanation of the most common metrics that can be used to evaluate the performance of different extraction methods (Section 3.2). Section 3.3 provides an overview of related research in which definition extraction has been addressed using a pattern-based approach. The remainder of the chapter describes my own approach and the results obtained with it. Section 3.4 introduces the pre-processing steps that have been taken to enrich documents with linguistic annotation and to convert them into a common format. In Section 3.5, the local grammars that have been designed on the basis of the patterns described in Chapter 2 are presented. The grammars consist of XPath-based regular expressions which are matched against elements in the input document by the XML transducer *Lxtransduce* (Section 3.6). The quantitative results obtained with the pattern-based approach are outlined in Section 3.7 while Section 3.8 focuses on the qualitative evaluation. The chapter concludes in Section 3.9 with a summary of the main findings regarding the use of a pattern-based approach for definition extraction.

3.2 EVALUATION METRICS

Several metrics can be used to describe and evaluate the performance of a pattern-based extraction method. The most common ones are recall (R), precision (P) and (variations of) F-score (F). Recall is (in the definition extraction context) defined as the number of definitions extracted divided by the total number of definitions in the data set. A high recall means that most of the definitions that are present in a text are detected. Precision is defined as the number of definitions extracted divided by the total number of sentences extracted. This measure thus gives an indication of the amount of correctly extracted definitions. The F-score is the harmonic mean of precision and recall and provides insight into the balance between these two metrics. In some situations, the recall is more important than the precision. If this is the case the F_n-score can be used. The F_n-score is an adapted version of the F-score in which more weight is assigned to recall. Depending on how important the recall is, the metric can be adjusted by using different values of n (Rijsbergen, 1979). The formulae for the different metrics are:

$$\mathrm{R} = \frac{\text{definitions extracted}}{\text{total number of definitions}} \tag{3.1}$$

$$\mathrm{P} = \frac{\text{definitions extracted}}{\text{sentences extracted}} \tag{3.2}$$

$$\mathrm{F} = \frac{2 \times \mathrm{P} \times \mathrm{R}}{(\mathrm{P} + \mathrm{R})} \tag{3.3}$$

$$\mathrm{F}_n = \frac{(1 + n^2) \times \mathrm{P} \times \mathrm{R}}{(n^2 \times \mathrm{P} + \mathrm{R})} \tag{3.4}$$

For all metrics, the scores can have any value between 0 and 1, where 1 indicates a perfect result and 0 the worst result possible. A recall of 1 thus means that all definitions are retrieved. When the precision is 1, this means that all extracted sentences are definitions. An F-score of 1 means that the set of sentences extracted exactly matches the manually annotated definitions whereas a low F-score means that the correlation between the manually annotated definitions and the automatically extracted sentences is low.

3.3 STATE OF THE ART

This section presents an overview of related research in which a pattern-based approach has been used for definition extraction in the context of question answering, dictionary building and ontology building. The last part of the section describes the work that has been carried out within the glossary creation context.

Question answering Definition extraction is a relevant task in the question answering (QA) domain since definitions can be used to answer certain types of questions. The question types that can be addressed with definition extraction techniques are the 'What is' (e.g. *What is an ace?*) and 'Who is' (e.g. *Who is Roger Federer?*) questions. They are called definition questions (Voorhees, 2002). A traditional QA system consists of three modules. In the first module, the question is processed to find the term to be defined. The term is assumed to be expressed explicitly in the question sentence. Questions that need more inference to identify the term are generally not considered. In the second module, documents are searched for fragments containing this term. As a last step, the possible answers are ranked on the basis of properties defined by the developer. Since only the second and third steps of the question answering process are relevant for the definition extraction task, the focus in this overview is on these two steps.

Joho and Sanderson (2000); Joho and Sanderson (2001) report on an experiment in which they try to identify hyponym-hypernym contexts to answer definition questions using nine regular expressions. These patterns include the copula verb 'to be', words and phrases like 'such as', 'which is', phrases between brackets, and appositions. Four of these patterns were proposed earlier by Hearst (1992). The fragments containing the query noun and one of the regular expressions are ranked on the basis of three properties. As a first property, the system employs the weights of the regular expressions. These weights reflect how often the expression is used in definitions and non-definitions. As a second property, it looks up the number of sentences in the document containing the term before it is used in the definition. The third property is the word count, which shows how many of the words that are

common across candidate answers are present in the sentence. To compute the word count, the first sentence of each document that contains the definition term is retrieved, and the 20 most frequent lemmas of these sentences are retained after applying a stop-list. The word count is the percentage of these words that are present in the sentence being ranked. The sentences are ranked using the weighted sum of the three attributes, after hand-tuning the weights of the sum on a training set. The method was evaluated with 50 definition questions and the top 600 documents that Google returned for each definition term. It returned a correct definition in the five top-ranked sentences in 66% of the questions.

Others who used hyponym-hypernym relations to answer definition questions are Prager et al. (2001, 2002), who retrieve the hypernyms from WordNet (Fellbaum, 1998). To identify the best hypernyms, it is counted how often they co-occur with the definition term in two-sentence passages of the document collection. The system was tested on the TREC-9 (2001) and TREC-10 (2002) datasets. In TREC-9 (2001), the system returned at least one correct response in the five most highly ranked two-sentence passages for 83.3% of the definition questions. In all these cases, the correct response was actually the highest ranked. In TREC-10 (2002), the definition questions more directly mirror real user queries, and as a consequence the percentage of correct answers dropped to 46%. In 44 of the 130 definition questions (33.8%), WordNet did not help at all, because it either did not contain the term for which a definition was sought, or none of its hypernyms were useful. This illustrates the main disadvantage of using a WordNet based approach. Furthermore, the approach is less suitable in our case where the term is not known in advance. However, WordNet could be used to filter non-definitions after the extraction of definition candidates.

Tanev (2004) presents a prototype of a QA system for Bulgarian, called Socrates. In addition to definition questions, this system addresses 'Where' and 'When' questions. To answer the definition questions, the system selects fragments containing the term within one of six manually created lexico-syntactic patterns. The resources investigated are Google snippets and encyclopaedic resources. The six patterns include fragments containing the copula verb 'to be', four pat-

terns in which a punctuation character is used (',', ':', '-' and '(') and complex noun phrases containing a common noun and a proper noun (e.g. 'The tennis player Roger Federer'). The system was evaluated on 100 questions from the TREC 2001 question collection, which have been translated into Bulgarian. To evaluate the results the Mean Reciprocal Rank (MRR) has been used instead of the precision and recall metrics. This means that the top five ranked answers are considered to assign a score and the position within this top five determines how good an answer is. The formula used is $\frac{1}{position}$, which means that the score is 1 if the correct answer is ranked first and 0.2 if it is the fifth fragment. The total score from all the questions is then divided by the number of questions, thus obtaining a MRR between 0 and 1. The MRR of Socrates on the definition questions was 0.447. These results show that the integration of encyclopaedias in a QA system can significantly increase its performance and that pattern based QA can be quite efficient.

Saggion (2004) addresses the problem of discriminating between definitional and non-definitional text passages about a particular definiendum in vast text collections in the context of answering definition questions. To this end, he developed a method that integrates a restricted number of lexico-syntactic patterns and terms that co-occur with the definiendum in on-line sources for both passage selection and definition extraction. In order to find good definitions, a collection of patterns like 'DEFINIENDUM is a' and 'DEFINIENDUM consists of' for identification and extraction of definiens is used. The problem they identified is that there are many ways in which definitions are conveyed in natural language, which makes it difficult to come up with a full set of lexico-syntactic patterns to detect all definitions. To make matters more complex, patterns are usually ambiguous, matching non-definitional contexts, as well as definitional ones. As a solution, Saggion (2004) uses external sources to mine knowledge which consists of terms that co-occur with the definiendum before trying to define it using the given text collection. This knowledge is used for definition identification and extraction. The approach of Saggion resulted in an F_5-measure (recall five times more important than precision) of 0.236 on the TREC QA 2003 competition. They do not give precision and recall separately, but state that one of the reasons for their low recall is

that in many cases, answers could not be extracted because the definition patterns and filters were far too restrictive to cover them.

Han et al. (2007) propose answer extraction and ranking strategies for definitional question answering using linguistic features and definition terminology. The extraction of answer candidates is based on five syntactic patterns. A passage expansion technique based on simple anaphora resolution is used to retrieve more informative sentences. In order to rank the phrases, several pieces of evidence are used to compare the different candidates. These are redundancy, local term statistics, external definitions and definition terminology. The most relevant features seem to be the external definitions and the definition terminology. The recall of the system evaluated on TREC 2004 is 0.3124, which is comparable to the other systems that participated.

Dictionary building Muresan and Klavans (2002) developed a rule-based system, DEFINDER (Klavans and Muresan, 2000), to identify and extract definitions and the terms they define from on-line consumer health literature. The system combines shallow natural language processing with deep grammatical analysis. DEFINDER aims at automatic constructing dictionaries from text to be used in summarizations of technical articles. The definitions should provide the explanation of technical terms in lay language. Their corpus consisted of consumer-oriented medical articles, which have a very typical text structure: almost 60% of the definitions are introduced by a limited set of text markers ('–', '()'), the other 40% being identified by more complex linguistic phenomena (anaphora, apposition, conjoined definitions). The DEFINDER system is based on two main functional modules, the first one uses cue-phrases (e.g. 'is the term for', 'is defined as', 'is called') and text markers in conjunction with a finite state grammar to extract definitions. The other module, which is used to achieve higher accuracy, is a grammar analysis module based on a statistical parser (Charniak, 2000) in order to account for several linguistic phenomena used for definition writing (e.g appositions, relative clauses, anaphora). The evaluation of the system was done by comparing the results against human performance. Four subjects (not trained in the medical domain) were provided with a set of nine articles (length of articles was limited to two

pages), and asked to annotate the definitions and their headwords in text. The system was evaluated against a gold-standard that consisted of a set of 53 definitions marked by at least 3 out of the 4 subjects. The interpretation of the results was more difficult than expected, given that there was no agreement among users regarding the question "What is a definition?", even though they were provided with a set of instructions and sample definitions. The DEFINDER system was able to detect 40 out of 53 definitions (75.47% recall) with a precision of 86.95%.

Ontology building Research on definition extraction has been carried out in the area of ontology building as well. Definitions are mainly used in this area to detect relations between concepts. For example, within the German HyTex project (Storrer and Wellinghof, 2006), 19 verbs that typically appear in definitions ('definitor verbs') were distinguished and search patterns have been specified based on the valency frames of these definitor verbs in order to extract definitions. Definition detection approaches developed in the context of question-answering-tasks (such as the method proposed by Saggion (2004)) are definiendum-centered, i.e. they search for definitions with a given term. This approach, in contrast, is connector-centered, i.e. they search for verbs that typically appear in definitions with the aim of finding the complete list of all definitions in a corpus independently of the defined terms. The corpus consisted of 100,000 words in 20 technical documents and contained 174 definitions. Depending on the verb, the results obtained strongly varied; the precision was between 9 and 100% (with an average of 31%) and the recall was between 20 and 100% (with an average of 70%). In addition to extracting the definitions, semantic relations have been extracted from them. Even though this information has been employed for the automatic generation of hypertext views that support both reading and browsing of technical documents, the same technique could be used to update and enlarge existing formalized ontologies as well.

Walter and Pinkal (2006) investigated the use of computational linguistic analysis techniques for information access and ontology learning within the legal domain with the goal to improve the quality of a text based ontology learning method. In their approach, they first applied 33 extraction rules based on connectors to the corpus to extract

definitions from parsed text. Precision using all 33 rules was 48.6% and increased when only using the best 18 rules to 75.2%. The issue of recall was not addressed. Adjective-noun-bigrams are often used as a basis in text based ontology extraction, since in many cases such bigrams contain two concepts and one relation: the nominal head represents one concept, while adjective and noun together represent another concept that is subordinate to the first one (e.g. *definition* and *nominal definition*). Walter and Pinkal (2006) extracted such adjective-noun-bigrams from definitions because they expected that the presence of concepts in definitions indicates that it is an important concept. They then compared the quality of noun-adjective bigrams extracted from the whole corpus and the noun-adjective bigrams contained within the definitions. Using this method increased the precision, whereas the recall dropped dramatically.

Glossary creation Velardi et al. (2008) proposed a three step method for the creation of domain glossaries. Just as in question answering, their system starts with a list of terms that should end up in the glossary. This makes the approach different from ours in which the terms are not known in advance. The document collection from which the definitions are extracted consists of the web. This is another difference with our approach, in which one document at a time is sent to the system which then aims at detecting as many definitions as possible within that specific document. In the first step, Velardi et al. (2008) use Google to retrieve sentences that match a restricted set of regular expressions observed in definitions for each of the terms ('*t* is a', '*t* is an', '*t* are the', '*t* defines', '*t* refers to', '*t* concerns', '*t* is the', and '*t* is any'). For each query, the first five pages are saved. A consequence of using the web as a corpus is that it is not necessary to include many types of patterns, since the web contains a lot of content whereas the glossary only needs to contain one definition for each term. For example, entering their regular expressions in Google returns 11.9 million results for the term 'XML' and 7.8 million for the term 'Excel'. However, this simple approach returns a lot of sentences that are non-definitions as well (e.g. 'Mastering Excel is a Critical Marketing Skill'). Therefore, a stylistic filter is applied to the returned candidates to match only sen-

tences expressed in terms of genus and differentia. To this end, a J48 decision tree algorithm has been trained on a data set of positive and negative examples from four domains. The combination of the regular expressions and the stylistic filter is similar to what we are doing in the pattern-based approach, as the grammars combine regular expressions and the structure of definitions (stylistic filter) to detect definitions.

There are two differences between the approach by Velardi et al. (2008) and our approach. These are the use of a more extensive set of regular expressions and the implementation of the stylistic filter within the grammars instead of applying it as a filtering step. The third step in the approach from Velardi et al. (2008) is the use of a domain filter to remove candidate definitions that are not pertinent to the domain of interest. The list of terms used as the input to the glossary extraction algorithm is analyzed to learn a probabilistic model of the domain, which assigns a probability to each single word that occurs in the terminology. Although the domain filter is quite relevant in the context of Velardi et al. (2008) where the definitions are extracted from the web, it is less important in our context where the extraction method is document-based.

Within the LT4eL project, a pattern-based approach has been used to develop a semi-automatic tool for the creation of glossaries in eight languages (Bulgarian, Czech, Dutch, German, English, Polish, Portuguese, and Romanian). The distinction of definition types based on the connector as proposed in Westerhout and Monachesi (2007a) was adopted within the complete project and has thus been used in all languages (Przepiórkowski et al., 2007a; Iftene et al., 2007; Del Gaudio and Branco, 2007).

Przepiórkowski et al. (2007a) present work on definition extraction in the Slavic languages that were represented in the project, that is, Bulgarian, Czech and Polish. The three grammars for these languages show varying degrees of sophistication, with a small Bulgarian grammar (8 rules in a 2.5 kB file), a larger Polish grammar (34 rules in a 11 kB file) and a sophisticated Czech grammar (147 rules in a 28 kB file). The bigger the grammar, the more patterns are covered with its rules. As a consequence, the best recall scores were obtained with the sophisticated Czech grammar (recall of 46%) and the small Bulgarian gram-

mar delivered the worst results (8.9%). Although the recall improved when more rules were added, the precision remained comparable for the three languages and varied between 22.3 and 22.5%.

Iftene et al. (2007) describe experiments on the extraction of Romanian definitions. The Romanian grammar has been evaluated at two levels, namely at the sentence and at the token level. At the sentence level, a sentence is called a definition when at least one token is marked as a definition whereas at the token level for each individual token – word or punctuation character – it is tested whether or not there is a match between the automatically and manually annotated definitions. The recall when evaluating the Romanian grammar at the sentence level was 100% for all types, which would mean that all definitions have been found. However, at the token level this score was considerably lower and varied between 10.2 and 33.6 % depending on the type of definition pattern. No explanation has been provided for this huge difference between the two evaluation methods.

In their work on the extraction of Portuguese definitions, Del Gaudio and Branco (2007) distinguish three definition types, namely *is*, *verb* and *punctuation* definitions. They use a rule-based grammar to extract these types and evaluate the grammar on unseen data. The recall scores for the *is* and *verb* patterns are similar (66% and 65%) whereas for the *punctuation* patterns it is considerably lower (47%). The precision is quite low for all three types (32% for *is*, 14% for *verb* and 28% for *punctuation*).

3.4 PRE-PROCESSING THE DOCUMENTS

The previous section described how definition extraction has been addressed with a pattern-based approach in the past. There are pattern-based methods that are entirely based on lexical phrases (e.g. 'is a') and methods that are based on a combination of lexical and syntactic information (e.g. pattern of definiendum). In my approach, the second method has been adopted. Since lexico-syntactic patterns are used as part of this approach, it is necessary to pre-process the documents as a first step to add the linguistic information.

The conversion of documents is necessary for two reasons. First,

one common format is necessary to enable tools like the glossary candidate detector to interpret the content of documents. The second reason is that the texts need to be linguistically annotated, since the pattern-based extraction of Dutch definitions is largely based on linguistic annotation. The conversion of the documents consists of two main phases. In the first phase, the original documents (i.e. PDF, HTML, DOC) are converted into a common XML format. The structural and layout information from the documents are stored in the XML. The glossary candidate detector uses the structural information of the XML to match sentences and tokens. In the second phase, the documents are annotated with three types of linguistic information: part-of-speech (POS) tags, lemmas, and morpho-syntactic information. The documents are split into sentences before the linguistic annotation is carried out.

The conversion and annotation process thus consists of different steps. Since all tools make some errors, the final documents can contain several types of errors. Most problems are caused by the sentence splitter and the linguistic annotator. The sentence splitter cannot always detect sentence boundaries properly in sentences that do not end with a period, such as headers and list items. The linguistic annotation tool does not always assign the correct categories to words. The problems are discussed in more detail in section 3.4.2.3.

3.4.1 DOCUMENT FORMAT

The learning objects of the LT4eL corpus have different document formats, namely PDF (49%), HTML (22%) and DOC (29%). To process the documents with different tools (e.g. search, keyword extraction, definition extraction), it is necessary that all files have the same structure. Therefore, all corpus documents have been converted into one common XML format. The aim of the conversion process is to add linguistic information and to integrate this information with the structural and layout information in a final XML format conforming to the LT4ELAna DTD (cf. Appendix A). The LT4ELAna DTD is an enriched version of the XCES DTD for linguistically annotated corpora (Ide and Suderman, 2002). It provides the possibility to encode parts-of-speech, morphosyntactic features and lemmas in addition to storing text and layout

information. The extension of the XCES DTD makes it possible to mark definitions and keywords. This information is used by the LT4eL keyword extractor and the glossary candidate detector presented in this thesis.

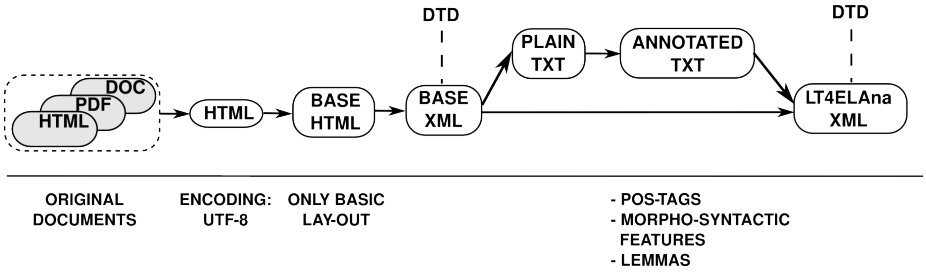

Figure 3.1 *Conversion of documents from original format to annotated XML format*

Figure 3.1 illustrates the conversion process from the original input document (i.e. PDF, HTML, DOC) into the final XML output document. As can be seen from this diagram, the conversion process consists of several steps. First, the original formats are converted into HTML with UTF-8 encoding. In this format, all information from the original files with respect to layout is preserved. These HTML files are converted to a basic form of HTML (called 'base HTML') to simplify the document and to preserve only the most important layout information (e.g. bold, headers, links). All other layout information is removed from the file. In the next step, the base HTML is converted into base XML, which contains the same information as the base HTML, but in an XML format. In order to enable linguistic annotation, the base XML are converted to plain text, since the linguistic annotation tools that have been used work on plain text. As a last step, the base XML file and the linguistically annotated plain text file are merged into the final XML file, called LT4eLAna XML. The tools for keyword extraction and definition detection are based on this format.

In Figure 3.2, an example sentence in LT4ELAna XML format is presented. This example illustrates how the different types of attributes are used. The most important elements for the glossary candidate detector are the sentence (`<s>`) and token elements (`<tok>`). Since the grammars are matching within the sentences at the token level, the most relevant

```
<s id="s150">
  ...
  <tok id="t2254" class="word" base="het" ctag="Art"
    msd="bep,onzijd,neut">het</tok>
  <tok id="t2255" class="word" rend="b"
    base="eLearning-actieplan" ctag="N" msd="soort,ev,neut">
    eLearning-actieplan</tok>
  <tok id="t2256" class="punc" rend="b" base="." ctag="Punc"
    msd="punt">.</tok>
</s>
```

Figure 3.2 *Part of a sentence in LT4ELAna format*

information for our purposes is encoded in the <tok>-element. The *id* attribute is a unique identifier for each word, the *base* attribute contains the lemma of the word, the *ctag* attribute contains the part-of-speech tag and the *msd* attribute gives the morpho-syntactic information. The layout information is stored in the *rend* attribute. In the example, a bold typeface has been used in the original documents for the word 'eLearning-actieplan'.

3.4.2 LINGUISTIC ANNOTATION

3.4.2.1 Part-of-speech tagging

The pattern-based method is largely based on part-of-speech (PoS) tags which have been embedded in the regular expressions. In PoS tagging, the task is to assign the most appropriate morphosyntactic category to each token in a sentence taking the context into account. A PoS tag thus gives information about the function of a token in the context in which it occurs. Some tokens for the same word can get different tags depending on the context in which they are used. The input to a PoS tagging program is a sequence of tokens and the output is a single best tag for each token.

The Memory Based Tagger (MBT) developed by Daelemans et al. (1996b) has been used for annotating the Dutch documents with part-of-speech information and morphosyntactic features. In the memory-based approach used by MBT, a set of cases is kept in memory, each of them consisting of a word or lexical representation of it together with

	Accuracy	Percentage
Known	97.1	94.5
Unknown	71.6	5.5
Total	95.7	100.0

Table 3.1 *Tagging accuracy on known and unknown words*

the preceding and following context (two positions to the left and two positions to the right), and the corresponding category for that word in that specific context. A new sentence is tagged by selecting for each word in the sentence and its context the most similar case(s) in memory, and extrapolating the category of the word from these nearest neighbours. MBT is based on the assumption that two factors determine the part-of-speech tag of a word, namely its lexical probability and its contextual probability.

The MBT tagger-generator architecture has been applied to the written part of the Eindhoven corpus, which was tagged using the WOTAN tagset (Berghmans, 1995). The tagger was generated on the basis of the 610,806 first words of the tagged corpus (27,651 sentences). The performance of the resulting tagger has been tested on the 100,000 last words (5,763 sentences) of the 710,806 word Eindhoven corpus (Daelemans et al., 1996a). The accuracy of the tagger is considerably better on known words than on unknown words (cf. Table 3.1).

3.4.2.2 Lemmatization

Lemmatization is the "process of classifying together all the identical or related forms of a word under a common headword" (Kennedy, 1998). The pairing of words with their base forms is subject to three constraints:

1. The base form must be an independently existing word;
2. Pairing is performed on a word-by-word basis;
3. Each word form receives exactly one base form.

Lemmatization is thus more than only stemming, that is, removing or ignoring affixes. For example, when a stemmer is instructed

that removing '-ing' results in the lemma of a verb, it will not be able to lemmatize words like 'tasting' correctly, because the lemma is not 'tast', but 'taste'. A lemmatizer should recognize that words like 'better' and 'best' are inflected forms of 'good' and that the inflected verb forms 'went', 'gone' and 'goes' belong to the lemma 'go'. Lemmatizers should be able to deal with all such complexities and irregularities of a language (Kennedy, 1998).

The lemmatization task can be performed automatically. For the lemmatization of the LT4eL corpus, the Dutch lemmatizer MBLEM developed by Bosch and Daelemans (1999) has been used. This memory-based lemmatizer has been trained on CELEX data and was also used for the lemmatization of the Spoken Dutch Corpus (Eynde et al., 2000).

3.4.2.3 Annotation problems

The tools for tokenization and linguistic annotation have been trained and tested on an amount of documents and performed quite well on these texts. However, within the LT4eL corpus the content of the texts and the format in which they are available causes two problems that might not occur in other situations. These problems are related to the tokenization and linguistic annotation of the texts and have direct consequences for definition extraction.

A first problem is caused by the fact that in the computing and eLearning domains a relatively large number of English words are used. Since the tagger does not recognize foreign words, it does not know to which category they belong. Unknown words are either tagged with the tag 'Misc(vreemd)', which means that the tagger does not know the word, or the tag 'N(eigen,ev,neut)', which means that the tagger considers it to be the single form ('ev') of a proper ('eigen') noun ('N'). When the class 'Misc' is assigned to a word, it is not possible to distinguish which tag it should have. It can be anything, from an adjective to a verb. This makes it more complicated to describe definition patterns in terms of the linguistic information of words and word groups. When a word is tagged as 'N(eigen,ev,neut)' incorrectly, it also causes problems, as this often is not the correct tag for the word. In this case, the error makes it more complicated as well to describe the definition patterns on the basis of the linguistic information. Later in this chapter

it is described how the grammars deal with such errors.

A second problem is related to the tokenizer and is caused by the way the original documents are formatted. The end of headers, enumeration items, and table of contents entries generally do not contain a punctuation character to indicate that the sentence ends. Since there is no punctuation mark, the tokenizer encounters difficulties in detecting sentence boundaries. As a consequence, sentences that should actually be separated are linked together. This makes it more complicated for the linguistic annotation tools to assign the correct tags to all words.

3.5 GRAMMARS OF DEFINITIONS

According to Barnbrook (2002), the language of dictionary definitions can be considered a sublanguage. Although the patterns observed in glossary definitions are more diverse than the patterns from dictionary definitions, the investigation from Chapter 2 reveals that glossary definition sentences form a restricted subset of a language as well. To describe the different patterns contained in a sublanguage, a local grammar can be used. The link between local grammars and sublanguages is made explicit by Hunston and Sinclair (2000), who state that "It is possible, then, to see the items described by local grammars as small (but not insignificant) sub-languages, and sub-language descriptions as extended local grammars."[1] Since the language of the Dutch LT4eL definitions can be considered a sublanguage, local grammars can be used to describe its patterns.

The Dutch local grammar used initially consisted of rules matching as many definition patterns as possible, regardless of the definition type. Results were then evaluated on the basis of the performance of the complete grammar and it was not possible to split them for the four definition types. For pragmatic reasons, this grammar has been split in four smaller grammars, one for each of the four types. This makes it easier to investigate the performance of the grammar for each type of definitions (Westerhout and Monachesi, 2007a).

All grammars can be divided in five parts, each of them containing its own type of regular expressions. Part A uses part-of-speech infor-

[1] Hunston and Sinclair (2000). A local grammar of evaluation, p. 77

mation to define individual words. The second part, part B, contains regular expressions defining phrases. Part C and D match the definiendum and connector and in the last part (part E) all pieces are put together and the complete definitions are analyzed. Although there are separate grammars for the four types, there is a large overlap between the individual grammars with respect to the regular expressions to match words or phrases (part A, B and C).

To make it easier to refer to the regular expressions defined in the grammars, a code has been assigned to each expression based on four aspects.

Example	Description	Possible values
B	refer to part of the grammar	A, B, C, D, E
G	refer to definition type	I, V, Pu, Pr, G
1	number within rule 'BG'	1, 2, 3, ..., *n*
b	sub rule of rule 'BG1' (optional)	a, b, c, ..., z

Table 3.2 *The code elements to refer to regular expressions illustrated on the basis of regular expression 'BG1b'*

The codes for the regular expressions consist of three or four parts (Table 3.2). The first letter indicates which part of the grammar the regular expression appears in (A-E). The second code element refers to the definition type to which the expression applies (G(eneral), I(s), V(erb), Pu(nctuation), Pr(onoun)), where the 'General' category contains the ones that are included in each of the four grammars. Since each grammar part can contain more than one expression, the third code element can be used to number them (1, 2, 3). Finally, it is possible that a number of expressions are closely related to each other. These are indicated with small case letters at the end of the code (a, b, c).

To describe the patterns defined by the grammar a semi-formal notation based on Rebeyrolle and Tanguy (2001) has been used. Table 3.3 shows the meaning of the different abbreviations in this notation. Section 3.5.1 uses the semi-formal notation to define the general regular expressions that are included in each of the four grammars while Section 3.5.2–3.5.5 describe the ones that are different depending on the definition type.

Abbreviation	Meaning
X	any phrase or word
X_1	first token of the sentence
X_{CAP}	word beginning with a capital letter
«...»	word between the angle brackets is a verb (phrase), different inflections can be used of it (e.g. for 'to be' it can be 'is', 'are', 'was', etc.)
|	or, the scope of the rule is one word to the left and one word to the right. If it refers to more than one word, the string is put between parentheses
*	the string after which it directly appears can appear any number of times
[...]	string between square brackets is optional
`...'	single quotes around a string express that it is not part of the semi-formal notation

Table 3.3 *Semi-formal notation for the description of definition patterns*

3.5.1 GENERAL REGULAR EXPRESSIONS

This section contains the regular expressions that have been included in all the grammars.

3.5.1.1 Part A: Defining tokens

In Part A, the individual words (e.g. verbs, nouns, adverbs) and words between quotes using part-of-speech information and morpho-syntactic information are defined. In addition, it contains two regular expressions to define the beginning of a sentence and one to define the end of a sentence. Since little variation is observed between the four definition types at the word level, the majority of expressions in this part are general. Only the *pronoun* definitions contain some specific rules for matching pronouns.

AG1: Quotes Quotes can be put around words or phrases to emphasize them. Both single and double quotes are captured. The regular expression of this rule is simply "`|"".

AG2a and AG2b: Adjectives Adjectives can be used to modify a noun or pronoun by describing, identifying, or quantifying words. In Dutch, it often precedes the noun it modifies. Examples of adjectives are words like *fair*, *blue*, and *big*. A broad and a narrow regular expression related to the adjectives has been implemented. The narrow one is used in the noun phrase of the definiendum while the broad one is included in the remaining noun phrases.

The narrow adjective expression (AG2a) includes only adjectives and participles (e.g. *spoken*, *hidden*). In the second one (AG2b), a broader view on adjectives leads to the inclusion of ordinal numbers (e.g. *first*, *tenth*) and indefinite pronouns (e.g. *each*, *both*, *much*, *any*) in addition to the elements of AG2a. Although the indefinite pronouns are related more closely to articles than to adjectives, they have nevertheless been treated as adjectives in the grammar. The third regular expression on adjectives, AG2c, describes a conjunction of adjectives.

Since the adjectives are sometimes surrounded by quotes or parentheses, these have been included as optional elements in the regular expressions. Summarizing, the regular expressions for the adjectives are:

1. AG2a: "`[AG1|'('] Adj | V(participle) [AG1|')']`"
2. AG2b: "`[AG1|'('] Adj | V(participle) | Num(ordinal) | Adv | Pron(indefinite) [AG1|')']`"
3. AG2c: "`AG2b ',' | 'en' AG2b`"

AG3: Adverbs An adverb is a word that modifies another word or phrase (apart from nouns), such as verbs, adjectives, clauses, sentences, and other adverbs. In Dutch, there is a distinction between adverbs and adverbial adjectives. Adverbial adjectives are adjectives that are used as an adverb. For example, in the sentence '*The car drives fast.*', the word *fast* would be an adverbial adjective. In the grammars, both adverbs and adverbial adjectives are included in the regular expression to match adverbs: "`Adv|Adj(adverbial)`".

AG4: Articles The Dutch language distinguishes two definite articles ('de' (*the*) and 'het' (*the*)) and one indefinite article ('een' (*a*)). The word

'het' is also commonly used as a personal pronoun (third person singular, neutral form) or as a neutral indefinite pronoun (*it*). Since the tagger tagged the definite article 'het' several times incorrectly as a neutral indefinite pronoun instead of an indefinite article, I took the pragmatic decision to include the neutral indefinite pronoun 'het' into the regular expression to define articles. The regular expression defining articles is "`Art|Het(Pron(indefinite))`".

AG5a, AG5b and AG5c: Nouns Within nouns, the two main types are common and proper nouns. The infinitive form of a verb can be used as a noun also in some cases. Whereas the MBT tagger tags such words as verbs, they are considered nouns in the regular expression to match nouns. A third type of words that can be nouns are symbols (e.g. 'at'-sign (`@`) and dash (-)) and foreign words, which are a subtype within the category miscellaneous (Misc) according to the MBT tagger. The regular expression to match nouns includes all these types in AG5a: "`N|V(infinitive)|Misc(foreign)|Misc(symbol)`". The regular expression AG5b matches with quotes around the noun ("`AG1 AG5a AG1`") and in AG5c both options (AG5a and AG5b) are included in one expression: "`AG5a|AG5b`".

AG6: Prepositions The MBT tagger distinguishes four types of adpositions all of which are tagged as 'Prep'. The most common one in the LT4eL corpus is the preposition (89.3%), which is an adposition that is used before a noun phrase. Adpositions coming after the element to which they belong, called postpositions, are less common in Dutch (only 0.01% of the adpositions). The third type is the preposition 'te' (*to*), which is used before an infinitive. This adposition type constitutes 10.5% of the adpositions. The last type are the circumpositions in which a preposition and postposition are combined, such as 'door...heen' (*throughout*) or 'naar ... toe' (*to*), where 'door' and 'naar' are prepositions and 'heen' and 'toe' postpositions. Only 0.2% of the adpositions are of this type. The grammar only includes the prepositions in its regular expression: "`Prep(voor)`".

AG7: Tokens in sentence This regular expression defines any token except for the period. It is used in regular expressions to match a definition until the period indicates that the end of the sentence is reached.

AG8: Beginning of the sentence In most situations, the beginning of a sentence is the first word within an <s>-element. However, the sentence splitter tool did not detect sentence boundaries properly in all situations. Problematic cases are sentences that do not end with a period because they appear in headers, titles or tables. During the conversion process, information about the structure of documents has been preserved as much as possible using a tool developed within the LT4eL project, but this tool did not manage to include all layout information correctly. As a consequence, the end of the sentence is not detected in some cases which results in two sentences being considered as one sentence. For this reason, an extra regular expression is included in addition to the standard situation where definitions begin at the first position of a sentence. It defines capitalized words as possible startings points of a sentence. Based on the observation that capitalized words directly preceded by prepositions are not at the beginning of a sentence, the regular expression includes this as a restriction. Regular expression AG8 combines the three possible sentence starts in one regular expression. Priority is given to the first <s>-element of the sentence. If no definition begins at this position, the rule looks successively at capitalized words and quotes as possible starting points. Summarizing, the regular expression for matching the beginning of a sentence is "X_1|([AG1] $X_C AP$)".

3.5.1.2 Part B: Defining phrases

Part B contains regular expressions defining phrases (e.g. noun phrases, prepositional phrases). Instead of using a chunker to detect chunks, the chunks are defined in the grammar to make it easier to put restrictions on them. Just as in part A, most of the expressions in this part are type-independent and are included in each of the four grammars. These are an expression to define noun phrases (BG1), one to define preposition phrases (BG2) and one to define fragments between brack-

ets (BG3). Part B of the *is* and *pronoun* grammars contains an additional rule to match appositions.

BG1: Noun phrase The regular expression to define noun phrases integrates a number of expressions defined in part A. A noun phrase optionally begins with an article (AG4). After that, one or more adjectives (AG2b or AG2c) can be used which should be followed by one or more nouns (AG5c). The nouns are the only obligatory part of the noun phrase. Optionally, after the noun(s) the phrase can be extended to a conjunction of noun phrases. The only obligatory part is thus the noun: "`[AG4] [AG2b|AG2c] AG5c [',|'en'|'of' [AG4] [AG2b|AG2c] AG5c]`".

BG2: Preposition phrase A preposition phrase is a sequential combination of a preposition (AG6) and a noun phrase (BG1), its regular expression is "`AG6 BG1`".

BG3: Fragment between brackets A phrase between brackets can occur at different places in a text. In the manually selected definitions, such phrases are observed many times directly before the connector. Since the bracket phrase can contain anything, the regular expression to define it has only two requirements: it has to begin with an opening bracket and then matches any token (AG7) until a closing bracket is found: "`'(' AG7* ')'`".

3.5.1.3 Part C: Defining definiendum

As described in Chapter 2, a typical definition contains at least three elements: definiendum, connector and definiens. In the grammar, two of these three elements are identified by individual regular expressions. Part C contains expressions to define definienda and in part D the connector is expressed. The definiendum is in most definitions a noun phrase used at the beginning of a sentence. Therefore, four regular expressions to describe different types of noun phrases beginning at the first sentence position are included in the grammar (CG1a-CG1d). The last expression of part C (CG2) combines expression CG1a-CG1d.

CG1a - CG1d: Four definiendum types The restriction that a definiendum must begin at the first sentence position makes it necessary to write a separate regular expression for each of the noun phrases that can occur at this position. In regular expression CG1a, the definiendum consists of one or more nouns of which the first one is used directly at the beginning of the sentence. Rule CG1b matches definienda beginning with an article at the first position of the sentence, optionally followed by an adjective and ending with at least one noun. In the third rule (CG1c), the definiendum begins with one or more adjectives followed by one or more nouns. The last definiendum type (CG1d) is a definiendum between quotes. The part between the quotes is a noun phrase containing at least one noun.

In semi-formal notation, the four regular expressions are:

1. CG1a: "AG5a$_{CAP}$ AG5c*"
2. CG1b: "AG4$_{CAP}$ [AG2a*] AG5c*"
3. CG1c: "AG2a$_{CAP}$ [AG2a*] AG5c*"
4. CG1d: "' [AG4] [AG2a*] AG5a* '"

CG2: Combining definiendum types Regular expression CG2 begins with one of the phrases of CG1. Optionally, after this a preposition phrase or conjunction can follow. The defined phrase is the definiendum part of the definition: "CG1a|CG1b|CG1c|CG1d [BG2] [*en* BG1]".

3.5.2 GRAMMAR FOR *is* DEFINITIONS

On top of the general regular expressions presented in the previous section, there are a number of expressions that are specific to the *is* definitions. The expression that defines *is* definitions is the most important rule of this grammar (EI1). Apart from this rule, the *is* grammar contains an expression for matching appositions (BI1), which are used only in *is* and *pronoun* definitions.

BI1: Defining appositions An example of an apposition in an *is* definition is the phrase 'ook wel dissertatie genoemd' in the following definition:

(9) Een proefschrift , ook wel dissertatie genoemd , is een boek
A 'proefschrift' , also dissertation called , is a book
geschreven door een promovendus met daarin een
written by a PhD candidate with in it a
wetenschappelijke verhandeling over een bepaald onderwerp
scientific treatise on a certain topic
.
.
'A 'proefschrift', also called a dissertation, is a book written by a PhD candidate containing a scientific treatise on a certain topic.'

An apposition, as defined by the grammar, is a phrase beginning with a comma followed by any number of words until a sequence of a comma and a verb is found: "`',' AG7* V ',`". This regular expression has been included in the *pronoun* grammar as well (BPr1).

EI1: Defining *is* definitions In this regular expression, three types of *is* definitions are described:

1. `In | Volgens BG1 « zijn » BG1 [AG3] BG1 AG7* '.'`
   `In | (According to) BG1 « to be » BG1 [AG3] BG1 AG7* '.'`
2. `CG2 « zijn » [AG3] BG1|BG2 AG7* '.'`
   `CG2 « to be » [AG3] BG1|BG2 AG7* '.'`
3. `AG7* « zijn » BG1 '.'`
   `AG7* « to be » BG1 '.'`

The first regular expression is used to identify definitions that describe who is responsible for that specific definition. This pattern is useful for glossary creation, since in some cases the context makes a certain definition more specific. Such specific definitions may not be contained in a dictionary. The second and third pattern are more common patterns of *is* definitions. The second and by far most common regular expression defines a sequence of a definiendum, connector and definiens. The definiens begins in most cases with a noun phrase or preposition phrase. In some of the definitions, the first word of the definiens is an adverb, such as 'dus', 'gewoon', or 'eigenlijk'. A number of adverbs observed in the patterns of the development corpus have therefore been included in this regular expression of the grammar. In the third pattern, the order of the definiendum and definiens are switched and the

sentence begins with the definiens and ends with the connector and definiendum.

3.5.3 GRAMMAR FOR *verb* DEFINITIONS

In addition to the general regular expressions from Section 3.5.1, the *verb* grammar contains four additional rules.

DV1: Verbal connector phrase at beginning of sentence The first verbal definition type involves sentences in which the definiendum is mentioned at the begin of a sentence. This definiendum is followed by a verbal connector phrase and the definiens. This regular expression has also been included in the *pronoun* grammar (DPr1). In total, this regular expression matches 36 types of verbal connector phrases using 27 different verbs or verb phrases. The regular expression for this rule is "`CG2 « verb phrase » AG7* '.'`".

DV2: Verbal connector phrase at end of sentence In the second type of *verb* definitions, the definiendum is mentioned at the end of the sentence. The grammar describes eight possible variations of this type using seven distinct verbs. The verbal connector phrase can be discontinuous in this case. The expression to match these definitions is "`AG7* « verbal phrase » BG1 [ « second part verbal phrase » ] '.'`".

DV3: Verbal connector after preposition phrase In the last type, the definiendum is contained in a preposition phrase after which the verbal connector phrase follows. Two pattern types can be distinguished:

1. `Bij | Met | Onder BG1 « verbal phrase » AG7* '.'`
   `By | With BG1 « verbal phrase » AG7* '.'`
2. `BG2 « kunnen | worden » BG1 « verbal phrase » AG7* '.'`
   `BG2 « can | are » BG1 « verbal phrase » AG7* '.'`

In the first type, the sentence begins with a preposition phrase that contains the definiendum after which the connector phrase and definiens follow. The other type begins with a preposition phrase (that does not contain the definiendum) that is followed by a copula verb. The

definiendum, verbal phrase and definiendum constitute the rest of the definition.

EV1: Defining *verb* definitions The three types described in DV1 - DV3 are combined in this regular expression: "`DV1 | DV2 | DV3`".

3.5.4 GRAMMAR FOR *punctuation* DEFINITIONS

On top of the general expressions presented in Section 3.5.1, the *punctuation* grammar contains an additional rule to match colon and bracket definitions (EPu1).

EPu1: Defining *punctuation* definitions The regular expressions to define the two types of punctuation patterns are:

1. `BG1 [ '(' AG7* ')' ] : AG7*`

2. `BG1 '('|'[' AG7* ')'|']'`

There are some restrictions on the first word after the colon and within the brackets. After the colon, capitalized words are not allowed based on the observation that these often are names of authors or other types of proper nouns. Within the brackets, it is for the same reason not allowed that the phrase begins with a capitalized word. In addition, imperative verbs are not allowed at this position as these are in our corpus mainly used to refer to another source. By far the most frequently observed imperative verb in this position is the verb 'Zie' (*see*).

In addition to imperative verbs and capitalized words, there are some other word types that are used only in non-definitions in this position. Since excluding all types that are observed would result in overfitting to the dataset, I decided that a word type had to account for at least 5% of the extracted non-definitions to be excluded by the grammar. This was only the case for capitalized words and the imperative verbs, all other phenomena occurred less frequently.

3.5.5 GRAMMAR FOR *pronoun* DEFINITIONS

On top of the regular expressions that are used in all grammars (Section 3.5.1), the *pronoun* grammar contains a number of additional ex-

pressions in part A, B, D and E of the grammar.

APr1a: Defining pronominal adverbs of the type 'waar...' In this expression, a number of pronominal adverbs of the type 'waar...' (*... which*) are matched, such as 'waarmee' (*with which*), 'waarin' (*in which*), 'waarbij' (*by which*) and 'waardoor' (*through which*). These are used as connector in one of the *pronoun* definition types. These words are matched with the regular expression "`'waarmee'|'waarin'|'waarbij'|'waardoor'`".

APr1b: Defining pronominal adverbs of the type 'hier...' and 'daar...' In this regular expression, a number of words of the type 'hier...' and 'daar...' are matched, namely 'hiermee' (*with this*), 'hierin' (*in here*), 'hierbij' (*herewith*), 'hierop' (*upon this*), 'daarmee' (*with that*), 'daarin' (*in it*). The MBT tagger tagged these words either as standard adverbs ('hiermee', 'hierin', 'hierbij', 'daarmee') or as pronominal adverbs ('hierop', 'daarin'). Since they are considered pronominal adverbs in this thesis, definitions in which they are used as connector are called *pronoun* definitions. The expression that has been used to match these tokens is "`'hiermee'|'hierop'| 'hierin'|'hierbij'|'daarin'`".

BPr1: Defining appositions This regular expression is the same as BI1: "`',' AG7* V ','`".

DPr1: Verbal connector phrase at beginning of sentence This regular expression resembles DV1. The only difference is that it begins with a demonstrative determiner (e.g. 'dit', 'dat') or pronoun (e.g. 'dit', 'zo'n', 'zulke'): `Pron(demonstrative) (AG5c) « verbal phrase » AG7* . .`

DPr2: Verbal connector phrase at end of sentence This is a restricted version of DV2. It is only different at the beginning of the definition: `Pron(demonstrative) (AG5c) « verbal phrase » BG1 [ « second part verbal phrase » ] .`

DPr3: 'dat wil zeggen' The Dutch phrase *dat wil zeggen* corresponds to the English phrase 'i.e.' and is often used as a connector between definiendum and definiens. It can be used as an abbreviation as well (*d.w.z.*). Because of the pronoun *dat* ('that'), sentences in which this phrase has been used as the connector have been classified as *pronoun* definitions. The expression to match this phrase is "`'dat wil zeggen'|'dit wil zeggen'|'d.w.z.'|'D.w.z.'|'dwz'`".

EPr1: Defining *pronoun* definitions In addition to the patterns described in DPr1, DPr2, and DPr3, there are two other *pronoun* patterns. EPr1 combines the five types of *pronoun* definitions:

1. `BG1 APr1a AG7* .`
2. `APr1b « verbal phrase » AG7* .`
3. `Pron (AG5c) « verbal phrase | zijn » BG1 AG7* .` (DPr1)
4. `Pron (AG5c) « verbal phrase » BG1 [ « second part verbal phrase » ] .` (DPr2)
5. `CG2 « dat wil zeggen » AG7* .` (DPr3)

The first pattern begins with a noun phrase that is followed by a relative pronoun of the type 'waar...'. The second definition pattern begins with 'hier...' or 'daar...'. The remaining three types have been described in DPr1, DPr2, and DPr3.

3.6 LXTRANSDUCE

The regular expressions presented in the previous section have been converted to *Lxtransduce* rules to make matching of definitions possible. *Lxtransduce* is a finite state XML transducer intended for use in natural language processing (NLP) applications (Tobin, 2005). XPath-based regular expressions are matched against elements in the input document. When a match is found, a corresponding rewrite is done. *Lxtransduce* can work with either text or XML documents as input formats. The corpus documents in which the definitions have to be annotated are XML files. The grammars are thus defined over XML elements and are therefore called 'XML-level grammars'.

Elements matched by a rule are replaced with the rule's rewrite, while for the unmatched elements the input remains unchanged. The

output XML files produced by *Lxtransduce* thus include annotated definitions in addition to the information that was already available in the input file. The structure of the XML documents has been described in 3.4.1. The definitions (<definingText>-element) are annotated within the sentence (<s>-element). Within the definitions, two other elements can be added: <markedTerm> to wrap the definiendum and <connector> to wrap the connector phrase. Example 10 shows in italics what is added to the original XML structure when a sentence is matched by the grammar:

(10)
```
<s id="s459">
  <definingText def_type1="is_def">
    <markedTerm>
      <tok msd="soort,mv,neut" ctag="N" base="grafiek" rend="b"
        id="t7365">Grafieken</tok>
    </markedTerm>
    <tok msd="hulpofkopp,ott,1of2of3,mv" ctag="V" base="zijn"
      id="t7366">zijn</tok>
    <tok msd="attr,stell,vervneut" ctag="Adj" base="grafisch"
      id="t7367">grafische</tok>
    <tok msd="soort,mv,neut" ctag="N" base="weergaves"
      id="t7368">weergaves</tok>
    <tok msd="voor" ctag="Prep" base="van" id="t7369">van</tok>
    <tok msd="soort,mv,neut" ctag="N" base="waarde"
      id="t7370">waarden</tok>
    <tok msd="voor" ctag="Prep" base="op" id="t7371">op</tok>
    <tok msd="bep,onzijd,neut" ctag="Art" base="het"
      id="t7372">het</tok>
    <tok msd="soort,ev,neut" ctag="N" base="werkblad"
       id="t7373">werkblad</tok>
    <tok msd="punt" ctag="Punc" base="."id="t7374">.</tok>
  </definingText>
</s>
```

The *Lxtransduce* grammar rules use three types of information contained in the attribute values of the <tok>-elements to formalize the regular expressions. These are the '`ctag`'-attribute (containing the PoS tag), the '`msd`'-attribute (containing the morpho-syntactic information), and the '`base`'-attribute (containing the lemma).

To understand how the *Lxtransduce* grammar rules are structured, the basic principles are explained in Table 3.4. Example 11 shows how these principles are used in the grammar rules. This example provides a simplified version of the rule to match *is* definitions. A more detailed description of the grammar file format can be found in the online man-

element	attribute	description
rule		refers to grammar rules
	name	name of the rule
	wrap	rewrite is wrapped in new element
	attrs	attributes of newly defined element
ref		refers to another rule defined earlier in the grammar file
	name	name of the rule to which it refers
	mult	number of matches of the rule: * (0 or more), + (1 or more), or ? (0 or 1)
query		matches a regular expression
	match	regular expression to be matched
	mult	see `<ref>`
constraint		imposes extra constraints on the match
	test	regular expression whose value must be true for the rule to match
first		matches the first token to match
best		matches the longest stretch of tokens to match
seq		matches the input if the child elements match in order
and		matches the input if all child elements match at the current point
not		matches the input if the child element does not match

Table 3.4 *Lxtransduce elements*

ual of *Lxtransduce* (Tobin, 2005).

(11)
```
<rule name="is_def" wrap="definingText" attrs="def_type1=
'is_def'">
  <best>
<!-- Na << zijn >> Nx|PP tok* .-->
    <seq>
      <ref name="simple_NPs"/>
      <query match="tok[@ctag='V' and @base='zijn' and
          @msd[starts-with(.,'hulpofkopp')]]" wrap="connector"/>
      <first>
        <ref name="noun_phrase"/>
        <seq>
          <ref name="prep_phrase"/>
          <ref name="noun_phrase"/>
        </seq>
      </first>
      <ref name="tok" mult="*"/>
      <query match="tok[@ctag='Punc' and @base='.']" mult="?"/>
    </seq>
<!-- In / Volgens NP << zijn >> Na (Adv) NP -->
    <seq> ... </seq>
```

```
<!-- tok* << zijn >> (een) Na . -->
    <seq> ... </seq>
  </best>
</rule>
```

At the top level, the 'rule' element provides the name of the *Lxtransduce* rule ('is_def'). The two other attributes tell what needs to be done when this expression is matched. The 'wrap'-attribute says that the phrase needs to be wrapped within a `<definingText>`-element and the 'attrs'-attribute indicates that an attribute should be included within the `<definingText>`-element that tells that the definition is an *is* definition. Within the `<best>`-element, a number of expressions is defined. The grammar tries to find the longest (best) match possible of these expressions. This can be either one of the three types of *is* definitions. In the example, only for the first type the pattern is defined. This should begin with a definiendum that is matched by a previously defined rule called 'simple_NPs'. After the definiendum, the connector verb *is* should be used and the connector has to be followed by either a noun phrase or a verb phrase. For the rest of the sentence, the lexico-syntactic structure is not considered and the rule simply matches any number of tokens (with the rule 'tok') until it comes to a full stop. If a sentence does not have this pattern, the other two types are tried. When a match is found, the definition is marked in the original XML document.

3.7 RESULTS

The regular expressions presented in Section 3.5 have been implemented in *Lxtransduce* grammars. The results obtained with the four grammars for the different definition types are analyzed in this section. The definitions that are not detected by the grammar are analyzed. After the discussion of the results for each type, some general observations are described and explained.

For the purpose of grammar development and testing, the LT4eL corpus has been divided into a development (75%) and a test part (25%) (Westerhout and Monachesi, 2008). The grammar has been evaluated on both the development and the test corpus. The difference between

#	Development	Test	Total
Documents	33	12	45
Words	311883	108319	420202
Sentences	23820	7732	31552
Definitions	473	130	603

Table 3.5 *General statistics of the definition corpus*

the results on the development and the test corpus give an indication of the robustness of the grammars. Table 3.5 shows the statistics for the development and test corpus.

The results have been evaluated in two ways. First, the results on the development and test corpus are compared to examine how well the grammars perform and how robust they are. In addition to the recall, precision and F-score, the F_2 score has been reported as well, since the focus of the pattern-based approach is on the recall. In addition to this, the patterns that are not extracted with the grammars are analyzed. The percentages are based on the definitions that have been marked manually.

3.7.1 RESULTS FOR *is* DEFINITIONS

corpus	definitions	R	P	F	F_2
devel	134	0.881	0.353	0.504	0.678
test	52	0.788	0.342	0.477	0.625
complete	186	0.855	0.350	0.497	0.664

Table 3.6 *Results of* is *grammar on development (devel), test and complete corpus*

Table 3.6 shows the results obtained with the 'to be' grammar. The results on development and test corpus are different with respect to recall and comparable with respect to precision where the grammar performs better on the development corpus.

Table 3.7 shows the problems for *is* definitions that are not matched. The majority of non-detected definitions (63.0%) are caused by tagger errors. As described in Section 3.4.2, the MBT tagger performs

	development		test	
	#	%	#	%
tagger error	9	6.7%	8	15.4%
sentence start	3	2.2%		
complex pattern	4	3.0%	2	3.8%
missing in grammar			1	1.9%
total non-matched	16	11.9%	11	21.2%
total matched	118	88.1%	41	78.8%

Table 3.7 *Non-matched definitions with* is *grammar*

considerably better on known words (97.6%) than on unknown words (71.6%). Definitions from the test corpus contain considerably more tagger errors than definitions from the development corpus (15.4% and 6.7% respectively). Error analysis reveals that both known and unknown words cause problems. For unknown words, two classes can be distinguished. Most errors of this type consist of assigning the wrong tag to names for things like programs, commands, and projects (e.g. *postscript*, *xpaint*, *tr*, *eWatch*). The second class to which a wrong tag is assigned frequently are words that are English from origin, such as *shell*, *link*, *mark-up*, *datagram* and *web-address*. Errors on known words have been made four times. Three times the wrong category has been assigned due to ambiguity of the word 'zijn', which can be either the infinitive form of *to be* or the possessive pronoun *his*. In all three cases, the connector verb has been tagged as a possessive pronoun. The other tagger error of this type, which occurs only once, is the conjunction 'of' (*or*) being tagged as a proper noun. It is not clear why this happened. When the definitions containing tagger errors are not considered, the recall improves to 93.2% (development) and 94.4% (test).

The other non-detected patterns either did not begin with a capital letter or had complicated sentence constructions. Only one non-detected definition had a pattern that should be added to the grammar. Appendix B contains a complete list with definitions that are not matched with the grammar grouped on the basis of the error types.

corpus	definitions	R	P	F	F_2
devel	150	0.807	0.467	0.592	0.704
test	51	0.686	0.368	0.479	0.585
complete	201	0.776	0.441	0.562	0.679

Table 3.8 *Results of verb grammar on development (devel), test and complete corpus*

	development		test	
	#	%	#	%
tagger errors	10	6.7%	3	5.9%
complex patterns	4	2.7%	2	3.8%
no verbal connector	13	8.7%	4	7.8%
other	2	1.3%	3	5.9%
missing in grammar			4	7.8%
total non-matched	29	19.3%	16	31.4%
total matched	121	80.7%	35	68.6%

Table 3.9 *Non-matched definitions with* verb *grammar*

3.7.2 RESULTS FOR *verb* DEFINITIONS

Table 3.8 shows the results for the *verb* definitions. The grammar is able to extract almost 70% of the definitions from the test corpus. Although this is considerably lower than the recall on the development corpus (80.7%) it is still quite high, which indicates that the grammar seems to cover most verbal patterns. The difference is not completely due to the grammar, since other factors play an important role also.

Table 3.9 provides an overview of the characteristics from the *verb* definitions not matched with the grammar. The main problem are definitions that actually do not have a definition pattern and do not contain a verb or verb phrase that can be used to identify definitions. In the development corpus, this accounted for 44.8% of the errors whereas for the test corpus this percentage was 25%. An example of such a sentence from the corpus is:

(12) Een vaste spatie voorkomt dat een regel tussen twee
A fixed space prevents that a line between two

woorden wordt afgebroken .
words is broken down .
'*A non-breaking space prevents that a line will be broken between two words.*'

It is very difficult to detect such sentences using a pattern-based approach and also with other approaches such patterns will probably be problematic to find. More investigation on contextual features of such definitions is necessary to find out whether other information can be used to detect them with a different approach. As far as I know, no research in this direction has been carried out yet.

Tagger errors are the second most frequent cause for non-detected patterns. The proportion is higher for the development and test corpus (34.5% versus 18.8%) and the errors are of the same type as the ones in the *is* definitions. For one quarter of the non-detected patterns from the test corpus, the errors involve definition patterns for which a regular expression should be added to the grammar, as these are missing definition patterns. For three of the four sentences, this would mean adding an extra connector phrase and for one sentence an additional phrase within the connector ('ook wel' (*also*)) would have to be allowed. These patterns account for 7.8% of the *verb* definitions. The errors that do not fit into one of the other categories are put in the group 'other'. These are problems like names of commands ("' x " permissie' ('*x' permission*)), a sentence ending with a semicolon, and a duplicated word ('voor voor' (*for for*)). Appendix B lists all *verb* definitions not matched with the grammar.

The results of the verb grammar differ largely for each connector phrase. The recall is above 0.5 for all phrases except for 'geven + NP' (*to provide + NP*), which can be used in a definition like '*Metadata provide a brief description of the material, such as author, publication date, keywords and, in the case of e-learning, teaching level*'. For this type, that has been used in the definition corpus four times, the recall is as low as 0.25. In particular, the variation of the precision scores is high and differs between 0.05 and 1. In Appendix E (Table E.1), the balance between definitions and non-definitions for each of the verbal connector phrases is shown. The most problematic phrase is constituted by the ambiguous verb 'leveren', which can be used in definitions in the meaning of *to*

corpus	definitions	R	P	F	F_2
devel	105	0.876	0.081	0.148	0.295
colon	69	0.942	0.120	0.213	0.398
bracket	30	0.900	0.045	0.086	0.188
other	6	0	0	0	0
test	7	0.857	0.018	0.035	0.083
colon	2	0.500	0.006	0.013	0.030
bracket	5	1.000	0.028	0.055	0.126
other					
complete	112	0.875	0.067	0.124	0.255
colon	71	0.930	0.095	0.172	0.236
bracket	35	0.914	0.041	0.079	0.114
other	6	0	0	0	0

Table 3.10 *Results of* punctuation *grammar on development (devel), test and complete corpus*

provide, but also has other senses, namely *to furnish*, *to purvey*, *to supply*, and *to deliver*. In some other phrases the low precision is partly due to the fact that the phrase is ambiguous as well. This is, for example, the case for the phrase 'bevatten' (precision of 0.14), which can mean *to comprise* or *to include*, but also means *to understand* or *to realize*.

3.7.3 RESULTS FOR *punctuation* DEFINITIONS

As described in Section 2.3.3, there are four types of punctuation connectors and thus also four types of punctuation patterns. The grammar focuses only on the detection of the two most common types. These are definitions in which a colon is used as the connector and definitions in which either definiendum or definiens is enclosed within brackets. The comma and dash types are not included in the grammar. Since both the colon and the bracket can be used for many purposes in addition to definitions, the precision for this type is very low. From Table 3.10 one can see that on the complete corpus the grammar performs quite differently on colon and bracket definitions with respect to precision, which is much lower for the bracket definitions.

The small number of *punctuation* definitions in the test corpus makes it difficult to compare the performance on development and test cor-

	development		test	
	#	%	#	%
tagger errors	3	2.9%		
problematic connector	6	5.8%		
complex patterns	2	1.9%	1	14.3%
other	2	1.9%		
total non-matched	13	12.6%	1	14.3%
total matched	90	87.4%	6	85.7%

Table 3.11 *Non-matched definitions with* punctuation *grammar*

pus. In five of the seven *punctuation* definitions from the test corpus, a bracket is used as the connector and in the other two cases a colon is used. In the non-detected pattern, the connector is a colon.

Table 3.11 shows the division of errors for the non-detected patterns. As a consequence of including only the two most frequently used connectors, the six definitions in which another connector is used are not detected. Almost half of the errors are caused by this problem (five times a comma, one time a dash). Another problematic issue is constituted by sentences in which connector and definiendum appear at the end of the sentence instead of at the beginning. These are often not recognized or only matched partially. The partial matches are caused by the design of the grammar, which is developed in such a way that it first searches for definitions in which the definiendum is at the beginning of the definition, followed by a colon or bracket and the definiens. Because the performance is evaluated on sentence level, these definitions are nevertheless classified as extracted. Again, an overview of all non-detected definitions can be found in Appendix B.

3.7.4 RESULTS FOR *pronoun* DEFINITIONS

As described in Section 2.3.4, the term *pronoun* definitions is an umbrella term for a number of patterns that all have in common that a pronoun is used to connect definiendum and definiens. However, there are several ways in which the pronouns are used and as a consequence there is a variety of pronoun definitions. This variety and the low frequency for each of the different types makes generalization difficult.

	#	R	P	F	#	R	P	F
		development				**test**		
this	30	0.767	0.144	0.242	8	0.500	0.133	0.211
this + np	12	0.500	0.500	0.500	2	1.000	1.000	1.000
here...	6	1.000	0.231	0.375	2	0.000	0.000	0.000
there...	16	0.875	0.035	0.067	1	1.000	0.007	0.014
that is	15	1.000	0.341	0.508	6	0.833	0.556	0.667
other	2	0.000	0.000	0.000	1	0.000	0.000	0.000
all	81	0.790	0.100	0.177	20	0.600	0.065	0.118
		complete						
this	38	0.711	0.142	0.237				
this + np	14	0.571	0.571	0.571				
here...	8	0.750	0.194	0.308				
there...	17	0.882	0.028	0.054				
that is	21	0.952	0.377	0.541				
other	3	0.000	0.000	0.000				
all	101	0.752	0.092	0.164				

Table 3.12 *Results of* pronoun *grammar on development, test and complete corpus*

To see how the grammar performs on the different types, the results in Table 3.12 are split according to five *pronoun* definition types.

The results show that the recall differs for each *pronoun* type. For the 'where' type, it is very low whereas for the 'dit - definiens' and 'dat wil zeggen' types it is much better in both the development and test corpus.

Evaluation of the results for the distinct pronoun types also reveals that the differences between the patterns have consequences for the de-

	development		test	
	#	%	#	%
tagger errors	7	8.6%		
complex patterns	3	3.7%	3	15.0%
no connector	7	8.6%	5	25.0%
total non-matched	17	21.0%	8	40.0%
total matched	64	79.0%	12	60.0%

Table 3.13 *Non-matched definitions with* pronoun *grammar*

	R	P	F	F_2
is	0.855	0.350	0.497	0.664
verb	0.776	0.441	0.562	0.674
punctuation	0.875	0.067	0.124	0.255
pronoun	0.752	0.092	0.164	0.309
all	0.813	0.158	0.264	0.444

Table 3.14 *Summarizing the results of the pattern-based approach*

gree to which they can be extracted using a pattern-based approach. The 'dat wil zeggen' type appears to be the easiest type to extract. From the results, the 'dit - definiens' type seems to be a problematic type to extract. However, if one would leave out the patterns that are not detected due to tagger errors, the recall score for the complete corpus would increase from 0.571 to 0.929. Three definitions have a pattern that is different from the five patterns described in the grammar. The decision to leave these out is justified by the fact that the patterns of these three sentences are quite complex and cannot be matched with general rules.

3.7.5 OVERALL RESULTS

Table 3.14 presents a summary of the results obtained with the pattern-based approach. The first general observation is that for all types recall is considerably higher than precision. Since the aim was to obtain the highest recall possible, this result is consistent with our expectations. The precision scores are especially low for the *punctuation* and *pronoun* definitions.

Recall The recall for all types is above 75%, despite the large number of different patterns that are observed in the data. The overall score is even 6% higher, which means that more than 80% of the definitions are detected within the texts. The reasons for patterns not being detected can be divided into three main groups. More specifically, these are tagger errors, problems related to the connectors and syntactically complex sentence constructions.

The tagger used, the Tilburg Memory Based Parser, has a reported performance comparable to other state-of-the-art taggers. However, all taggers make errors, especially when used on data different from their training data. It is thus inevitable that this happens in our corpus as well. To solve the first problem, the tagger needs to be improved. To reduce the number of problems related to the connectors, a solution might be to investigate the definition patterns in a larger corpus. The restricted number of definitions makes it sometimes difficult to define a general pattern and some of the connectors appeared only in the test data. The grammar could be made more stable if more patterns would be examined. However, this solution involves a lot of manual effort.

Precision The aim of the pattern-based approach is to match as many definitions as possible. Since the patterns of definitions are quite diverse, the regular expressions had to be quite general to obtain a high recall. An obvious side effect of using such general expressions is that the precision decreased. An example which clearly illustrates the negative influence of using general rules on the precision is the type of article used in the definiendum of *is* definitions. The majority of these definitions either contains no article or contains an indefinite article (87%) whereas only 13% contains a definite article. Using a restricted rule to match only definienda in which an indefinite article or no article is used would result in eliminating 44% of incorrectly extracted definitions. However, it would also cause a loss of 13% of the *is* definitions, which is not acceptable because of the primary focus on obtaining the highest recall possible with the grammar.

One of the reasons for the low precision is the inclusion of a rule that was used because of pragmatic reasons, namely errors of the sentence splitter tool. During the conversion from the original document (PDF, DOC or HTML) to XML, all sentences are split automatically and each sentence is wrapped within an <s>-element. The grammar searches for definitions at the level of sentences. During the sentence splitting process not all sentences have been split correctly, and the resulting errors have not been corrected manually. Such splitting errors typically occur when sentences do not end with a full stop, which is often the case in headers, list items or table entries. As a consequence of these errors,

it is not enough to state in the grammar that a sentence begins at the beginning of an <s>-element. To solve this issue the grammar states that each word beginning with a capital letter can indicate the beginning of a sentence. Although including this rule increases recall, at the same time it has a negative effect on the precision, since it also matches proper nouns, which are not necessarily used at the beginning of a sentence.

A lot of non-definitions have been eliminated with the pattern-based approach. However, it is still necessary to adopt another approach on top of the pattern-based one to increase the precision. To this end, a machine learning component is used in succession to the pattern-based approach as a filtering or refining step. By attempting to learn what differentiates a definition from a non-definition on the basis of a set of characteristics or features, it is possible to further improve the results obtained through the pattern-based approach. Features can be various, some based on linguistic information, whilst others based on different characteristics such as the importance of the word, the position within the document, and layout information. The details of the machine learning approach and the results obtained with it are discussed in the next chapters.

3.7.6 RESULTS WITH BASIC GRAMMARS

The grammars that have been presented in this chapter include regular expressions that are based on the connector phrases and the syntactic structure of definiendum and definiens. In order to investigate how important it is to include the syntactic patterns in addition to the connectors, a basic version for each of the grammars has been developed, which only match the connector phrases. These grammars has been evaluated on the same corpus as the sophisticated grammars.

Table 3.15 shows that the precision scores drop dramatically when the basic grammars are used while the recall increases. The improved recall scores can be explained by the fact that more complex patterns are matched and that tagger errors cause less problems. The decrease in precision shows us that the syntactic structure of the definiendum and the beginning of the definiens are of crucial importance when def-

	R	P	F	F_2
is	0.989	0.030	0.057	0.084
verb	0.837	0.088	0.159	0.218
punctuation	0.954	0.018	0.035	0.052
pronoun	0.762	0.034	0.065	0.093
all	0.892	0.033	0.064	0.092

Table 3.15 *Results with the grammars based only on the connector phrase*

initions need to be distinguished from non-definitions.

3.7.7 RESULTS FOR OTHER LANGUAGES

Within the LT4eL project, *Lxtransduce* grammars have been written for eight languages. In addition to Dutch, grammars were written for Bulgarian, Czech, English, German, Polish, Portuguese, and Romanian. The levels of sophistication of the grammars varied for each language and not all grammars addressed the four types described in this chapter.

	Is			Verb			Punctuation		
	R	P	F_2	R	P	F_2	R	P	F_2
Dutch	0.86	0.35	0.66	0.78	0.44	0.67	0.88	0.07	0.26
Bulgarian	0.67	0.28	0.52	0.68	0.24	0.49	0.25	0.10	0.19
Czech	0.48	0.30	0.43	0.25	0.09	0.18	0.62	0.21	0.44
English	0.58	0.17	0.39	0.32	0.34	0.32	0.12	0.33	0.14
German	0.55	0.37	0.50	0.20	0.32	0.22	0.17	0.02	0.06
Polish	0.74	0.22	0.50	0.40	0.13	0.28	0.64	0.04	0.17
Portuguese	0.66	0.32	0.54	0.65	0.14	0.38	0.47	0.28	0.41
Romanian	1.00	0.54	0.85	1.00	0.76	0.94	1.00	0.15	0.46

Table 3.16 *Results of grammars for all LT4eL languages*

Table 3.16 shows the results obtained for the eight languages with an *Lxtransduce* grammar. A sentence-based evaluation of the results has been used. This means that for each of the retrieved definitions it has been evaluated whether one or more of its tokens are part of a manually marked definition. If this is the case, the grammar is said to match that

sentence. The results for the *pronoun* definitions have not been included in the table, since only one grammar, apart from Dutch, addressed this type. Most grammars have been evaluated on the development corpus. For the Dutch and Portuguese grammars, a combination of a test and a development corpus has been used.

Although the same approach has been applied within each of the languages, there are large differences in the results. The Romanian results seem very good at first sight, but it is not clear how these results are obtained. The fact that the performance has been evaluated only on the development corpus makes the recall score of 1 less reliable. In addition, the results using a token based evaluation are quite different for the Romanian definitions. Both precision and recall drop dramatically: precision becomes between 0.003 and 0.064 and the recall decreases to values between 0.116 and 0.333. For the other languages, the sentence-based and token-based evaluation provided similar results.

The Dutch grammar clearly outperforms the grammars of the other languages. This is partly due to the fact that the development of most of the grammars stopped when the project ended. The language partners that proceeded to work on the glossary candidate detector after the project mainly focused on the use of machine learning techniques to improve on the grammar results (Kobyliński and Przepiórkowski, 2008; Borg et al., 2009; Del Gaudio and Branco, 2009).

3.8 QUALITATIVE EVALUATION

Within the LT4eL project, the grammars written for the glossary candidate detector (GCD) tool have been integrated in the Learning Management System (LMS) ILIAS. The usefulness of the tool and the results obtained with it have been tested for all languages. In the experiments, the test persons used the GCD functionality and other LT4eL functionalities (keyword extractor, semantic search) within the context of a scenario. After running the scenario, they had to fill in a number of statements on the different aspects of the system. The respondents specified their level of agreement to a statement on five-level Likert items (Likert, 1932).

The opinions on the quality of the output varied. 14 out of the 28 re-

sponses agreed on the statement that the terms proposed for definitions were suitable, 10 disagreed and 4 neither agreed nor disagreed. There was more agreement on the context provided with the terms: 17 out of 28 agreed that the context given for the term was accurate, whereas 7 disagreed and 4 neither agreed nor disagreed.

The users also provided additional feedback about the GCD functionality. There were comments regarding the usage of the tool, results obtained with the tool and the definitions extracted with it. With respect to the results, some test persons said that the quality of the results should be improved to make it a useful tool. Not everyone agreed, as other persons remarked that most of the retrieved definitions were correct.

A remark made by a test person considered the distinction between definitions and properties of definitions. Since the GCD aims at extracting broad definitions, which can in some cases be a description of the properties of a term, this distinction is not a big problem in the glossary creation context. To give an example, the definition 'Word is a program for editing texts' is not a definition in the sense that this description is unique to Word. Other text editors have exactly the same property. However, it is useful as a definition since it provides the learner with a description of what the program Word is intended for, which will be enough in most cases.

The responses to the statement 'The Definition finder is a useful tool' showed that tutors in general see the potential of such a tool in an eLearning context, as 21 out of 27 users agreed on this statement. The main reason to consider it a useful tool is that it assists them by providing a basis for a definition which they can eventually extend or adapt to their own needs.

3.9 CONCLUSIONS

This chapter described the use of a pattern-based approach to detect definitions. The XML transducer *Lxtransduce* has been used to match and annotate definition patterns in XML documents. The quantitative results obtained were first described for the language on which this thesis focuses, that is, Dutch. The general impression from this evalua-

tion is that the precision scores are not good enough, especially for the *punctuation* and *pronoun* definitions. The recall with our pattern-based approach is high.

Quantitative evaluation of the grammars written for the other languages of the LT4eL project reveals that the Dutch grammar outperforms all other grammars. This is probably due to the fact that for Dutch more efforts were invested in writing the grammars and defined as many as possible regular expressions of definitions. A lot of manual work needs to done to build an acceptable grammar due to the diversity of definition patterns. The qualitative evaluation of the LT4eL glossary candidate detector shows that users see the potential of such a tool within an eLearning context. However, there is disagreement with respect to the quality of the glossary candidate detector that has been implemented into ILIAS.

From the quantitative and qualitative evaluation it is clear that the precision scores need to be improved. Experiments to this end using machine learning techniques are presented in the next chapters. Since the recall scores obtained with the pattern-based approach is high, the grammar output constitutes a good basis for the further improvements. The machine learning approach is therefore used as a filtering step after the pattern-based extraction.

Observations always involve theory.

Edwin Hubble (1889–1953)

4 Machine learning

4.1 INTRODUCTION

Using a pattern-based methodology for the identification of definitions has one major drawback: a number of the defined patterns can be used in non-definitions as well. The result is that not only definitions have been extracted, but also a large number of sentences that have been incorrectly recognized as such. This is especially the case for the *punctuation* and *pronoun* definitions. In this chapter, a machine learning approach is presented that exploits several types of information in order to discriminate between definitions and non-definitions.

A machine learning process involves several steps. Decisions need to be taken regarding the data that will be used, the features to be investigated, and the selection of an appropriate classifier. To provide a basis for the choices that have been made in the design of the experiments, the first part of the chapter introduces the different components of a machine learning process. Furthermore, it describes for some of the components a number of options from which one can choose.

The use of appropriate features determines to a large extent how well the classifiers will perform. The pattern-based approach already revealed that definitions conform to a restricted number of lexico-syntactic structures. This chapter outlines a number of characteristics in addition to the patterns for which definitions and non-definitions behave differently. Some of these features are based on previous work in the area of definition extraction (cf. Blair-Goldensohn et al. (2004); Fahmi and Bouma (2006)), but I also identified several other characteristics that have not been investigated before. The features are related to

linguistic properties, connector properties, position properties, layout properties, and keyword properties.

The selection of a convenient classifier is based on several aspects. One element that is especially relevant concerns the balancedness of the data sets. Generally, the purpose of a classifier is to assign as many instances as possible to the correct class. As a consequence, many classifiers are not appropriate for imbalanced data sets in which the minority class contains the positive instances. Since some of the definition data sets that have been used are highly imbalanced, a classifier should be used that is able to handle imbalanced data sets. In my experiments, a classifier has been selected that addresses the imbalanced data set problem by using a balanced bagging version of the Random Forest classifier – the Balanced Random Forest classifier.

This chapter starts with an introduction to the field of machine learning (Section 4.2). This introduction addresses the different components that are involved in a machine learning process, such as the collection of data, selection of features and the choice of a classifier. The different components are linked to the definition classification task by describing a number of challenges and relevant aspects related to this task. Section 4.3 presents an overview of related research in which machine learning has been applied to definition extraction. The machine learning experiments that have been carried out on the data sets created with the pattern-based approach are presented in Section 4.4. The focus of this section is on the description of the features. Section 4.5 concludes the chapter by summarizing the design of the experiments.

4.2 MACHINE LEARNING COMPONENTS

Machine learning has been applied to a wide variety of tasks, such as recognizing spoken words, predicting recovery rates of pneumonia patients, detecting fraudulent use of credit cards, playing games, etc. (Mitchell, 1997). The classification of definitions falls within the category of classification tasks. In such tasks, a machine learning classifier is trained to assign unseen data to the correct category (or 'class') with the highest possible accuracy. To learn which classes new sentences belong to, an algorithm needs data. The amount of data depends on

the problem and the domain. A lot of examples are often needed for building a good classifier. The data set in the definition classification context consists of a collection of input sentences ('instances') that can be divided in two classes - definitions or non-definitions. Each instance is represented by a number of characteristics on the basis of which the algorithm learns to make predictions. These characteristics are called 'features'.

Kotsiantis (2007) presented a diagram that illustrates the components that are involved when applying machine learning to a problem (figure 4.1). The process starts with collecting a data set for a certain problem. The instances of this data set should be converted into a format the classifier can understand. This involves the selection of a subset of relevant features and removing redundant features if needed to reduce the dimensionality of the data (Yu and Liu, 2004). This is especially relevant when there are many features and the data set is big. In the algorithm selection component, an appropriate classifier is selected. To test and train this classifier the data set needs to be divided into training and test data.

The diagram shows that research into applying machine learning to a task is an iterative process. Backtracking to make changes in a previous component is generally part of the machine learning research. These changes can be very diverse: the data set can be enlarged, the feature set can be adapted, a different algorithm can be selected, or the parameters from the classifier can be tuned better. The rest of this section explains the different components in more detail and relates them to the definition classification problem.

4.2.1 IDENTIFICATION OF DATA

As described in Section 1.4, the interpretation of the term 'definition' can be more or less strict depending on the application in which definitions are employed. The application influences the desired outcome of the process as well: in question answering the aim generally is to obtain a ranked list with answer candidates, in dictionary building the most concise definition needs to be selected and in ontology building either the relation between genus and species has to be extracted or the

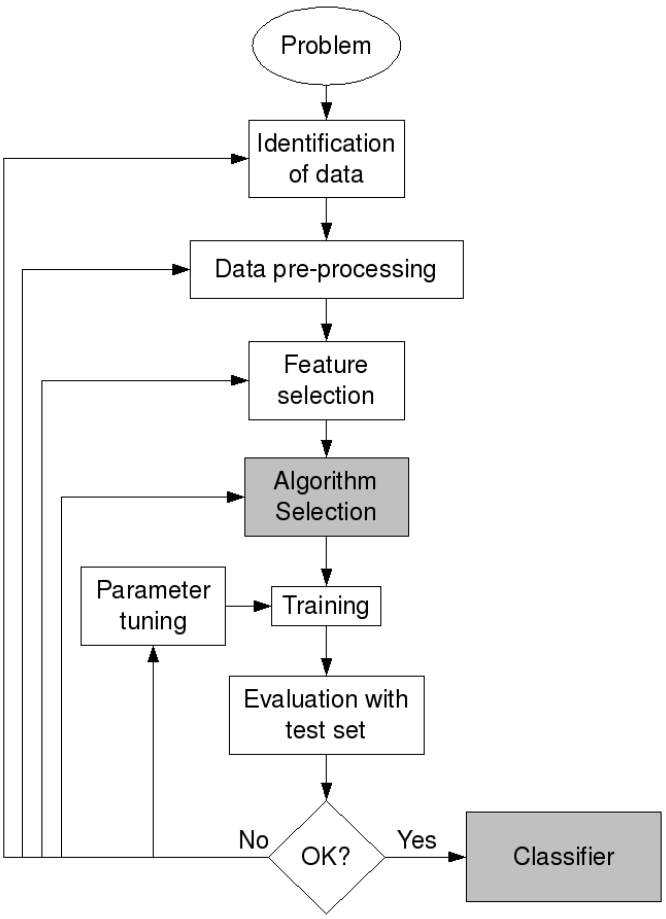

Figure 4.1 *The machine learning diagram adapted from Kotsiantis (2007)*

best definition for a given concept should be selected. Despite these differences, however, the shared task in all these applications is to distinguish definitions from non-definitions. The identification of appropriate data depends thus partly on the application and the interpretion of the term 'definition'. It is therefore not possible to simply take over a data set that has been collected for definition extraction purposes in a different context.

Several approaches can be adopted to collect a data set. The data set can be the product of a previous step (Fahmi and Bouma, 2006; Del Gaudio and Branco, 2009). Another way to acquire a data set is the manual collection of data. This involves the selection of appropriate content (documents, sentences, phrases) that are representative for the problem one aims to address. The data set should contain enough examples of the different classes that are distinguished. An additional task that is often involved when a data set is collected manually, is the labeling of the instances. Section 4.2.4 provides more details on the use of labeled or unlabeled data.

4.2.2 DATA PRE-PROCESSING

The collected data need to be converted into a format that can be understood by the machine learning algorithm. In this step, it is also decided how to deal with missing data. The format of the data depends on the program that is used. A common format that is understood by different machine learning programs is the comma-separated format (csv), in which the class is mentioned as the last value. This format can be read by machine learning applications like Weka (Witten and Frank, 2005) and TiMBL (Daelemans et al., 2007)). Weka uses its own arff format in addition to the comma-separated format. An arff (attribute-relation file format) file is an ASCII text file that describes a list of instances sharing a set of attributes. Conversion from csv to arff can be done automatically.

4.2.3 FEATURE SELECTION

In the definition classification task, the two classes are the definitions and the non-definitions. The ability to predict which class new sen-

tences belong to strongly depends on the features employed for training the classifier. The better these features distinguish definitions from non-definitions, the better the classifier will be at predicting the correct class of new data.

A very basic type of feature often used in text classification tasks are the *n*-grams (cf. Bekkerman and Allan (2003) and Tan et al. (2002)). This means that any consecutive combination of *n* items in a sequence is considered. The items included can be anything depending on the application (phonemes, syllables, letters, words, part-of-speech tags). Generally, *n*-grams of size 2 (*bigrams*) or 3 (*trigrams*) are used. An advantage of using *n*-grams as features is that little manual effort is involved in the selection of features. However, at the same time this naive way of feature selection can also be a disadvantage: it does not provide much insight in the characteristics of definitions. Another shortcoming is that they only consider immediately adjacent combinations in the text whereas other combinations can be relevant as well. In addition to using *n*-grams, feature selection can also be done on the basis of other aspects for which a different behaviour is observed in definitions and non-definitions.

Not all features are equally informative. The aim of the feature selection step is to define a set of features that lead to an optimal classifier performance. The selection of those features can be done either automatically or manually. The disadvantage of using bigrams does not provide insights in the performance of different combinations of features. If one wants to compare which types of information and combinations of features give good results, manual selection is necessary. Otherwise, the selection of the best features can be carried out automatically. Since the selection of appropriate features is restricted by the information that is available in the sentences, it will often be necessary to go back to the data identification phase to collect more information about the data.

4.2.4 ALGORITHM SELECTION

The selection of an appropriate algorithm to build the classifier takes place after the problem and data set have been defined. Since the prob-

lem that has to be addressed is the classification of definitions, a classification algorithm has to be used. To reduce the choice of potential classifiers even further, it is useful to make a distinction between three types of learning – these are supervised, unsupervised and semi-supervised learning (Chapelle et al., 2006). Each of these methods can be applied to classification tasks. In unsupervised learning, there is a set of n instances that have not been assigned to a class in advance. The goal of unsupervised learning is to find structures in the data that distinguish the different classes from each other. A common unsupervised method is clustering, where the aim is to find clusters of similar instances. In supervised learning, there is a set of n instances for which it is known to which class they belong. The purpose of supervised classifiers is to learn which class new instances belong to on the basis of the labeled training data. Semi-supervised learning has been designed to overcome the main disadvantages of supervised and unsupervised learning (Chapelle et al., 2006). In supervised learning, the labeling of instances is a time consuming and expensive task that involves a lot of manual effort from human annotators that are experts in the domain. Unlabeled data is much easier to collect, but starting without labeled data often demands too much from classification algorithms. In semi-supervised learning, a large amount of unlabeled data is combined with a restricted amount of labeled data to build better classifiers. In this way, the manual effort is reduced to a minimum whereas at the same time there is some knowledge from which the learning algorithm can start. In all related research that will be presented in Section 4.3, the data are labeled.

Another relevant aspect related to the classifier selection is the balancedness of the data set. A data set is said to be imbalanced (or skew) if the classification categories are not approximately equally represented (Chawla, 2005). Imbalance in class distribution is a common problem in real-world applications, ranging from predicting telecommunications equipment failure, text classification, learning word pronunciations, detecting fraud, to detecting oil spills from satellite images (Weiss, 2004). The data set in the definition extraction context can be either balanced or imbalanced. This depends on a number of factors, such as the definition types distinguished, the data collection process,

and the application for which definitions are extracted. For example, the application of question answering demands other output than the glossary creation context. Whereas in question answering the task of the classifier often is to order a number of possible answers to find the best answer, in glossary creation the classifier has to decide for each sentence whether it is a definition or a non-definition.

4.2.5 TRAINING AND TEST SET

Different techniques can be used to evaluate the accuracy of a classifier (Han and Kamber, 2006). The first option is to split the training set by using one part for training and another part for testing the performance ('holdout method'). A variation of this is random subsampling where the holdout method is repeated a specified number of times. It is possible that data occur in a subsample more than once.

In the second method, n-fold cross-validation, the original sample is partitioned into n subsamples of equal size. A single subsample of the n subsamples is retained as validation data for testing the model, and the remaining $n - 1$ subsamples are used as training data. The cross-validation process is then repeated n times (the 'folds'), and each of the n subsamples is thus used exactly once as validation data. The n results from the folds are averaged to produce a single estimation. A slightly different method is the stratified n-fold cross-validation, where the folds are stratified so that the class distribution of the subsamples is approximately the same as that in the initial data. The difference with the holdout method and random subsampling is that in n-fold cross-validation all instances are used exactly $n - 1$ times for training and one time for testing.

Leave-one-out cross-validation is a special case of n-fold cross-validation. As the name suggests, this method involves using a single observation from the original sample as test data, and the remaining observations as training data. This is repeated until each instance in the sample has been used once as test item. Leave-one-out cross-validation is computationally expensive because of the large number of times the training process is repeated.

		Predicted class	
		definition	non-definition
True class	definition	true positives (TP)	false negatives (FN)
	non-definition	false positives (FP)	true negatives (TN)

Table 4.1 *Confusion matrix showing results of classification*

4.2.6 EVALUATION METRICS

Precision, Recall and F-score The result of cross-validation is shown as a confusion matrix with $n \times n$ cells where n is the number of classes. The structure of the confusion matrix for the definition classification context is shown in Table 4.1. Precision and recall of both the definitions and non-definitions can be calculated on the basis of this contingency table (Formula (4.1)). Since in the context of definition classification the focus is on the classification of the definitions, the precision and recall are calculated for this class only. The F-score is calculated on the basis of the precision and recall in the same way as explained in Section 3.2.

$$\begin{aligned} P_{def} &= \frac{TP}{TP+FP} & P_{non-def} &= \frac{TN}{TN+FN} \\ R_{def} &= \frac{TP}{TP+FN} & R_{non-def} &= \frac{TN}{TN+FP} \end{aligned} \tag{4.1}$$

Accuracy The accuracy of a classifier is the proportion of instances that have been classified correctly without looking at how the classifier performs on the different classes. It can be calculated on the basis of the contingency table:

$$A = \frac{TP+TN}{TP+FP+FN+TN} \tag{4.2}$$

Although the accuracy is a common way to measure how well a classifier performs, the fact that it is based on the classification results as presented in the contingency table has an important drawback: the figures from the contingency table do not reflect the exact classification scores for the individual instances. As a consequence, a definition that is classified 'almost' as a definition according to this score (e.g. the

score is 0.45 where the threshold is 0.50) is dealt with in the same way as a definition having a very low score (e.g. the score is 0.05 with a threshold of 0.50). While evaluating a classifier, one would like to treat the incorrectly classified instances close to the threshold different from those that are far away from it.

AUC The AUC (area under the curve) score (Metz, 1978; Ling et al., 2003; Bradley, 1997) indicates how well a classifier performs on the basis of the scores obtained for each separate instance. It plots the true positive (TP) rate of a classifier in the range [0, 1] as a function of the false positive (FP) rate in the domain [0, 1]. For a fully random classification, the ROC (receiver operating characteristic) curve is a straight line connecting the origin to (1, 1). In this case, the classifier fails at providing good predictions. Any improvement over random classification results in an ROC curve at least partially above this straight line. The closer the curve is to the top left corner, the better it is. The ideal curve goes from (0,0) to (0,1) and then from (1,0) to (1,1). The AUC of this perfect curve is 1.0. The AUC is defined as the area under the ROC curve and is closely related to the ranking quality of the classification. An AUC score of 1 represents a perfect test whereas an area of 0.5 or lower represents a worthless test. The AUC score is reported to be less sensitive to imbalanced data than measures like precision, recall and F-score (Fawcett, 2004). In addition, the AUC score says something about the accuracy of the complete classifier and not specifically about the classification performance on one of the classes, which is the case with the precision, recall and F-score. A rough guide for classifying the accuracy of a classifier is:

0.5 – 0.6	no discrimination
0.6 – 0.7	poor
0.7 – 0.8	acceptable
0.8 – 0.9	excellent
0.9 – 1	outstanding

This section showed that classifiers can be evaluated in several ways. The AUC score can be reported instead of the accuracy to give a better

view on the quality of a classifier when imbalanced data sets are used. For the other metrics, the application determines the appropriateness of the method. In the case of question-answering, the performance can be evaluated by giving the proportion of questions that are answered correctly. Since the corpus used for this application is generally large and only one answer is needed for each question, the precision is more important in this context than the recall. When definitions are used to build dictionaries, it is relevant that the definition is detailed enough and that the precision is high. Since the corpus is large for this application as well, the recall is less important. When definitions need to be extracted in a glossary creation context, a number of aspects play a role. As has been emphasized earlier, it is especially important to have good recall scores, since the aim is to extract as many definitions as possible from each document. However, since the user needs to select definitions from the list with automatically extracted sentences, the precision needs to be acceptable as well.

4.2.7 PARAMETER TUNING

Many learning algorithms have parameters that can affect the outcome of learning. The parameter tuning component is concerned with finding the best settings for these parameters. The parameters to be tuned depend on the algorithm that is used. For example, a decision tree learner can have parameters that influence the amount of pruning while a k-nearest-neighbor classifier has one that sets the neighborhood size. When an algorithm is implemented to deal with balanced datasets, the parameters of this algorithm need to be tuned as well. Related to the tuning of the algorithm parameters, is the tuning of the features that are employed. For example, when n-grams are used, it can be useful to restrict the number of n-grams on the basis of information gain measures.

4.3 RELATED RESEARCH

The previous section introduced a number of aspects that must be considered when one wants to apply machine learning techniques to the

task of definition extraction. This section outlines the contributions from other researchers to this task. Since my approach aims to investigate which are the most relevant features for definition classification, the overview focuses on the features that have been used.

Question answering In the field of question answering, the DefScriber system (Blair-Goldensohn et al., 2004) focuses on answering definitional questions of the type "what is X?" by finding genus-species statements (sub type of the *is* definitions). They distinguish seven types of 'definitional predicates' (genus, species, target partition, cause, history, etymology and non-specific definitional fragments), of which DefScriber aims to find the genus, species and non-specific definitional fragments. A non-specific definitional fragment refers to information that is relevant in a detailed definition of the term. As can be seen from this enumeration, DefScriber does not focus on finding definition sentences but attempts to locate the seven relevant elements of a term that are then summarized into one definition.

The DefScriber system uses a machine learning and a pattern-based approach to detect definitions. In the machine learning step, a rule-learning classifier ('Ripper' (Cohen, 1995)) and a boosting-based categorization algorithm ('BoosTexter' (Schapire and Singer, 2000)) have been used. The machine learning features are based on term frequency, position of the sentence within the document, presence of punctuation (to prevent extraction of headers), and bag-of-words. With the machine learning, an accuracy of 81% was obtained. The second approach they used involves the use of manually constructed high-precision parse patterns to match genus-species type sentences. With this method, a precision of 96% was obtained while the recall score is not mentioned. The most relevant aspect from DefScriber for my work, is the set of features they used, which are mainly based on positional information. The coverage of the high precision patterns on our data is too small for the glossary creation context, since I focus on the recall instead of the precision.

Miliaraki and Androutsopoulos (2004) present a method (called DEFQA) to identify the best single-snippet answers to definition questions. As training data, 18,473 manually classified candidate answer snippets

that include a search term have been used of which only 3004 actually were definition snippets, which means that the data set is quite imbalanced. Their method integrates several types of features, some of which have been proposed by others. The features used include WordNet hypernym-hyponym relations (Prager et al., 2001, 2002), 13 lexical patterns (Joho and Sanderson, 2000; Joho and Sanderson, 2001)), and 200 word n-grams (1, 2 or 3) of phrases that occur directly before or after the definiendum. The classifier used is an SVM algorithm with a simple inner product (polynomial of first degree) kernel. For each question, a list of five answer snippets is shown (extracted from the web) on the basis of the highest definition scores. In this way, the approach gets around the imbalanced data problem. Evaluation shows that when all features are integrated the correct answer is contained in the first five snippets for 72.50% of the TREC-2000 questions and 84.67% of the TREC-2001 questions.

Miliaraki and Androutsopoulos (2004) deal with the imbalanced data set problem in a way that cannot be adopted in the glossary creation context since in the glossary context it is not known in advance which terms have to be included in the glossary. Whereas the purpose of Miliaraki and Androutsopoulos (2004) is to find a correct definiens for a given definiendum, in the glossary creation context complete definitions have to be detected. Since the definiendum is not known in advance, it is not possible to show the first n definition sentences the system returns. In the glossary context, the system has to make a distinction between definitions and non-definitions, which is a completely different task than ranking a number of possible answers to definition questions. As a consequence, the imbalanced data set problem needs to be addressed in a different way. Another important difference with their approach is the importance of recall. Miliaraki and Androutsopoulos (2004) focus on precision, because they are not interested in obtaining all definitions, a single correct one suffices. Therefore, they want to have high-precision indicators. In the glossary creation context where a glossary is extracted on the basis of one document, a good recall is crucial.

In a follow-up study on the work from Miliaraki and Androutsopoulos (2004), Androutsopoulos and Galanis (2005) propose a method to

label instances automatically, because manual classification is a laborious task. They exploit online encyclopaedias and dictionaries (from http://www.encyclopedia.com/ and Google's 'define:' feature) to tag training windows as definitions or non-definitions. Training instances in which the wording is very similar to that of the encyclopaedia definitions are tagged as definition windows (positive examples), while windows whose wording differs significantly from the encyclopaedia definitions are tagged as non-definitions (negative examples). The same classifier (SVM) and features are used as in Miliaraki and Androutsopoulos (2004). To evaluate the performance of the classifier, 81 target terms are used. Only the first snippet returned is considered, in contrast to Miliaraki and Androutsopoulos (2004) where five snippets were allowed. The classifier is able to provide a correct definition snippet for 58.02% of the target terms. Applying the method from Miliaraki and Androutsopoulos (2004) on the same data results in only 14.81% correct answers.

The training corpus composed automatically by Androutsopoulos and Galanis (2005) only considers patterns from encyclopaedia and dictionary definitions. As a consequence, a restricted variety of patterns is addressed. In their experiments, this is acceptable since their aim is to obtain a high precision while the recall is less important - one correct answer suffices. However, in the glossary creation application presented in this dissertation, the corpus from which the definitions need to be retrieved is smaller - one document instead of the web - and as a consequence, a larger diversity of patterns needs to be addressed.

Glossary creation Kobyliński and Przepiórkowski (2008) describe an approach in which the Balanced Random Forest classifier is used to extract definitions from Polish texts. They compare the results obtained with this approach to results obtained with experiments on the same data in which grammars were used (Przepiórkowski et al., 2007b) and to results of experiments with standard classifiers (Degórski et al., 2008). The best results are obtained with the approach designed for dealing with imbalanced data sets. The differences with my approach are that (1) they used either only machine learning or only a grammar and not a combination of the two, (2) they did not distinguish different defi-

nition types and (3) they only used relatively simple features, such as *n*-grams. They combined the four definition types into one data set and obtained a precision of 0.21, a recall of 0.69, and an F-score of 0.33.

Borg et al. (2009) applied Genetic Algorithms to the extraction of English *is* definitions. Their experiments focus on assigning weights to a set of features for the identification of such definitions. These weights act as a ranking mechanism for the classification of sentences, providing a level of certainty as to whether a sentence is actually a definition or a non-definition. With this approach, a precision of 0.62 and a recall of 0.52 is obtained for the extraction of *is* definitions using a set of features such as 'has keyword' and 'contains *is a*'. When they combined Genetic Algorithms with Genetic Programming, the precision was 1.0 with a recall of 0.51 and an F-score of 0.68.

Del Gaudio and Branco (2009) focus on the extraction of Portuguese *is* definitions. First, a simple grammar is used to extract all sentences in which the verb 'to be' is used as main verb. Because their corpus is heavily imbalanced and only 10 percent of the sentences are defintions, they investigate which sampling technique gives the best results and present results from experiments that seek to obtain optimal solutions for this problem. From their experiments, the SMOTE technique appears to be the best solution for dealing with imbalanced data sets in definition extraction. With this classifier, they obtained an F-score of 0.74 for the classification of *is* definitions. Since their experiments were carried out after my research was finished, the SMOTE technique has not been implemented into the classifier used for my experiments. The bagging approach that has been used in my experiments was missing in their comparison.

4.4 MACHINE LEARNING FOR GLOSSARY CREATION

As shown in the previous section, machine learning has been applied to the task of definition extraction for different purposes and in different ways. In my experiments, the focus is on improving the precision obtained with the pattern-based approach. To this end, experiments have been carried out in which a diversity of features are examined and compared. My investigation is restricted to document-based char-

acteristics, which means that external sources (like WordNet or online dictionaries) have not been considered. The description of my experiments is structured according to the diagram from Kotsiantis (2007) presented in Figure 4.1.

4.4.1 IDENTIFICATION OF DATA

The machine learning step has been preceded by the pattern-based extraction of definitions described in Chapter 3. To develop and evaluate this approach, 600 definitions were manually annotated. The sentences matched by the regular expressions contained in the grammar have been used as data sets for the machine learning experiments.

	Definitions	Non-definitions
is	35.0	65.0
verb	44.1	55.9
punctuation	6.7	93.3
pronoun	9.2	90.8
all	15.8	84.2

Table 4.2 *The proportions of definitions and non-definitions*

The precision scores obtained with the local grammars are quite different depending on the definition type. For the *punctuation* and *pronoun* types, the precision is low, which means that only a minority of the extracted sentences are definitions whereas the majority are non-definitions. These are typical examples of the imbalanced data sets that were introduced in Section 4.2.4. Table 4.2 shows the degree of imbalancedness for each of the data sets and reveals that it is especially high for the *punctuation* and *pronoun* sentences, in which less than 10 percent of the extracted sentences are definitions. For the *is* definitions, the data set is imbalanced as well. The low precision is the main reason for applying machine learning techniques on the results of the pattern-matching step.

4.4.2 DATA PRE-PROCESSING

The data set consists of sentences in LT4eLAna XML format (Appendix A) that have been enriched with some additional elements and attributes. The additional information is necessary to enable the inclusion of a variety of extra features in the machine learning data file that are not contained in the original LT4eLAna XML. An extra element has been added to encode the connector part of the definition (`<connector>`-element). At the level of the tokens (`<tok>`-element) a number of additional attributes are included. These are used to store position information, keyword information, and information on the type of connector phrase used. The position information (of definitions and definienda) has been derived from the original XML documents using an XSL script and has been included at the sentence level in the final XML data set. The keyword scores are calculated by the LT4eL keyword extractor (Lemnitzer et al., 2007) and have been merged in the final XML document using a Python script. The type of connector phrase is defined by the grammar and has been included as an attribute of the `<connector>`-element.

The machine learning package used for the experiments is the Weka tool (Witten and Frank, 2005). Since XML is not allowed as input format in Weka, the data sets have been converted into a different format. Formats supported by Weka are `csv` and `arff` (Witten and Frank, 2005). For the experiments, the XML data files have been converted to `arff` files. Appendix C contains an example `arff`-file. An XSL script has been used to create the arff files on the basis of the XML files .

4.4.3 FEATURE SELECTION

The data sets and literature have been investigated to identify a set of features that might be relevant for the classification of definitions. In this thesis, the focus is purely on information contained in the documents itself, without taking external resources (like WordNet or dictionaries) into account. The aim is to investigate which information in the document is relevant for distinguishing between definitions and non-definitions.

On the basis of the linguistic annotation of the documents, bigrams

of part of speech tags and morpho-syntactic information can be created and used for training. This is a naive way of collecting information on definitions, that simply considers any adjacent combination of two tags that appears in the sentences and compares whether there are differences between definitions and non-definitions in this respect.

The connector phrase was one of the key elements of the pattern-based approach. To investigate whether this information can be relevant in the machine learning experiments as well, several aspects related to the connector have been considered. Within the connector phrases that are typical for the four definition types a more fine-grained division is often possible (described in Section 2.3). For example, over 30 different connector phrases exist for the verbs. Another characteristic that has been investigated in this respect concerns the linguistic categories of the left and right context of the connector.

The pattern-based approach is based largely on linguistic information contained in definitions. However, analysis of the data reveals that there are at least two linguistic characteristics that were hard to address with a pattern-based approach, but can be included in the machine learning settings. The first problematic issue concerns the frequencies of the different patterns in definitions and non-definitions. The other problem involves the way in which the linguistic categories and syntactic phrases are combined in definitions and non-definitions. Compared to the linguistic bigrams, these two characteristics provide more insights in the use of specific linguistic properties. An additional advantage over the bigrams is that the amount of information needed for training and classification is much smaller. This increases the speed of the training and classification process.

Another feature of texts that may be relevant is related to the position of definition and definiendum within the document. Fahmi and Bouma (2006) used the position of sentences within Wikipedia articles as a feature. Over 60% of their definitions is the first sentence of the article. Although the position at first sight does not seem to play such a crucial role in the corpus documents, Blair-Goldensohn et al. (2004) showed that position information can be relevant in less structured documents as well.

To visualize that a word or phrase is relevant, it is possible to use

a different layout for these parts. Since definitions contain important information, I investigated whether specific layout features occur more often in definitions than in non-definitions.

The relevance of a phrase in a text can be expressed by its keyword score. The more important a term is within a text, the higher the keyword score. I examined whether this information can be used as a feature to distinguish definitions from non-definitions.

Summarizing, the features that have been used can be divided into six categories:

1. ***n*-gram properties**: these include bigrams of part-of-speech tags and morpho-syntactic information.

2. **Connector properties**: this category contains information on the connector. The connector itself is one of the features and the other features include information on the left and right context of the connector.

3. **Linguistic properties of definiendum and definiens**: in this category, linguistic information on specific words or phrases of the definiendum and definiens is included, such as, the type of noun used at the beginning of the definiens.

4. **Position properties**: these include several features that give information on the position within the document where the definition and definiendum are used.

5. **Layout properties**: this category deals with the layout of definitions.

6. **Keyword properties**: this category contains features on the importance of the definiendum within the document.

The remainder of this section presents an extensive comparison of the definitions and non-definitions with respect to these six properties. Since I wanted to discover to which extent the properties depend on the definition type, the comparisons have been made for each of the four definition types separately.

4.4.3.1 *n*-Gram properties

In many text classification tasks, *n*-grams have been used for predicting the correct class (cf. Bekkerman and Allan (2003); Tan et al. (2002)). For the classification of definitions, two types of bigrams based on linguistic information are considered - the Part of Speech tag (PoS) bigrams and the morpho-syntactic information (MSI) bigrams.

The MBT tagger that has been used for tagging the corpus documents distinguishes 9 PoS tags (Daelemans et al., 1996b): adjective (Adj), adverb (Adv), article (Art), conjunction (Conj), interjection (Int), noun (N), numeral (Num), preposition (Prep), pronoun (Pron), and verb (V). In addition, the tag miscellaneous (Misc) is used for unknown words and punctuation (Punc) is used for punctuation characters. Within the tags, a more fine-grained categorization results in a large number of sub categories. For nouns, for example, plural and singular forms are distinguished and a division is made in proper and common nouns. Such fine-grained categorization information is included in the morpho-syntactic information (MSI) attribute. Combining PoS and MSI information results in 215 different word types used in the corpus of which 175 have been used in the machine learning data sets.

Words	PoS	MSI
Een_nominalisatie	Art_N	Art(onbep,zijdofonzijd,neut)_N(soort,ev,neut)
nominalisatie_is	N_V	N(soort,ev,neut)_V(hulpofkopp,ott,3,ev)
is_een	V_Art	V(hulpofkopp,ott,3,ev)_Art(onbep,zijdofonzijd,neut)
een_woord	Art_N	Art(onbep,zijdofonzijd,neut)_N(soort,ev,neut)
woord_dat	N_Pron	N(soort,ev,neut)_Pron(betr,neut,zelfst)
dat_...	Pron_...	Pron(betr,neut,zelfst)_...(...)
..._...	..._...	..._...
..._.	..._Punc	..._Punc(punt)

Table 4.3 *An example of a bigram sequence of PoS tags (PoS) or PoS tags and morpho-syntactic information (MSI)*

Table 4.3 shows an example bigram sequence for the two bigram types. It shows that for all sentences, each combination of two adjacent words is extracted until the end of the sentence is reached. The bigrams thus do not contain information on the position within the sentence.

Whereas word bigrams have been used for the classification of defi-

nitions by several researchers (cf. Miliaraki and Androutsopoulos (2004); Androutsopoulos and Galanis (2005); Fahmi and Bouma (2006); Westerhout and Monachesi (2008)), less research has been carried out on the use of PoS and MSI bigrams. Only within the scope of the LT4eL project experiments started in which this information is investigated (cf. Kobyliński and Przepiórkowski (2008); Degórski et al. (2008); Westerhout (2009a); Del Gaudio and Branco (2009)). In the experiments discussed in this thesis, both PoS bigrams and MSI bigrams have been used. Experiments in which a combination of *n*-grams and the structure of definitions has been taken into account are presented in Westerhout (2009a).

bigram	definitions	non-definitions
is definitions		
Art(onbep,zijdofonzijd,neut)_N(soort,ev,neut)	4.49	2.24
V(hulpofkopp,ott,3,ev)_Art(onbep,zijdofonzijd,neut)	3.11	1.58
verb definitions		
Art(bep,zijdofmv,neut)_N(soort,ev,neut)	2.73	3.87
Art(onbep,zijdofonzijd,neut)_N(soort,ev,neut)	2.48	1.95
punctuation definitions		
N(soort,ev,neut)_Punc(komma)	2.62	1.51
N(soort,ev,neut)_Prep(voor)	3.51	2.25
pronoun definitions		
N(soort,ev,neut)_Adv(gew,geenfunc,stell,onverv)	0.81	1.95
Prep(voor)_Art(bep,zijdofmv,neut)	2.51	2.05

Table 4.4 *The use of MSI bigrams in definitions and non-definitions (in %)*

The proportions of bigrams have been compared for the definition and non-definition classes. This comparison reveals that there are a number of differences between the two classes, especially for the MSI bigrams. Table 4.4 shows that the differences depend on the definition type. Appendix D provides the top ten PoS and MSI bigrams for the definitions and non-definitions for each data set.

In the experiments, the PoS and MSI bigrams of each sentence are used as features to find out to which extent they are suitable for distinguishing definitions from non-definitions. The settings in which the bigrams have been used are referred to as the 'bigram settings'.

4.4.3.2 Connector properties

Definitions are distinguished on the basis of the connector. Since the corpus definitions contain four types of connectors, they are categorized in four groups. In the *is* definitions the connector is always the same, but in the other categories a variety of connector phrases can be used. Within the *verb* definitions, 36 different words or phrases are used to connect definiendum and definiens. In the *punctuation* definitions, two types of connectors are identified and in the pronoun definitions eight types can be distinguished.

Several pattern-based approaches have implemented a number of phrases (like 'is a', 'consist of', 'defines') that connect definiendum and definiens (cf. (Muresan and Klavans, 2002; Velardi et al., 2008)). A disadvantage is that these patterns are generally related to a restricted set of connectors and do not cover all definitions. On the basis of the pattern-based approach, an additional drawback has been discovered: connector phrases can often be used in non-definitions as well. In machine learning experiments, connector information is sometimes included implicitly in the bigrams (Androutsopoulos and Galanis, 2005; Fahmi and Bouma, 2006). The connector patterns have been used explicitly as well. In their genetic algorithm experiments, Borg et al. (2009) used the presence of the phrase 'is a' as a binary feature. Androutsopoulos and Galanis (2005) indicated whether the definitions contain one of the manually constructed patterns from Joho and Sanderson (2001) and Miliaraki and Androutsopoulos (2004).

Whereas all previous researchers (cf. Velardi et al. (2008); Borg et al. (2009); Androutsopoulos and Galanis (2005)) dealt with the connector phrase as one fragment (e.g. 'is a'), the connector and its context are addressed separately in my experiments. The first connector feature looks purely at the connector phrase without taking any context into account. The additional four connector features contain information about the linguistic categories of the words that appear directly to the left and right of the connector phrase. More specifically, these are the PoS tag of the word to the left, the PoS tag of the word to the right, the morpho-syntactic information of the word to the left, and the morpho-syntactic information of the word to the right.

The largest diversity of connector patterns is observed in the *verb*

connector	definitions	non-definitions
use to / for	19.23	11.68
consist of	10.9	10.66
mean	7.69	7.11
call	7.05	2.03
contain	5.13	24.87
define as	5.13	0
stands for	4.49	0.51

Table 4.5 *Verbal connector phrases in the* verb *data set*

definitions. Table 4.5 shows that in this data set there are several differences between the definitions and non-definitions. For example, the phrases 'call' and 'define as' are used considerably more in definitions whereas the verb 'contain' generally refers to non-definitions. Appendix E provides a complete overview of the different connector phrases that have been used in the *verb*, *punctuation* and *pronoun* definitions.

Summarizing, in the experiments five features related to the connector information have been included:

- Connector phrase
- PoS tag of the word to the left of the connector phrase
- PoS tag of the word to the right of the connector phrase
- morpho-syntactic information of the word to the left of the connector phrase
- morpho-syntactic information of the word to the left of the connector phrase

The setting in which these five features are combined is called the 'connector setting'.

4.4.3.3 Linguistic properties of definiendum and definiens

The linguistic features are used to investigate how linguistic information differs in definitions and non-definitions. There are at least two im-

portant linguistic characteristics that have not been taken into account within the pattern-based approach and can be addressed with machine learning. The first problematic issue concerns the frequencies of the different patterns in definitions and non-definitions. For example, in the *is* data set, a definiendum with an indefinite article is more common in definitions (24.5%), but it can be used in non-definitions as well (8.1%). The other problem involves the way in which the linguistic categories and syntactic phrases are combined in definitions and non-definitions. The patterns either allow or prohibit combinations but do not make a distinction between common and uncommon combinations.

The linguisic features that have been investigated can be divided into features related to the definiendum and features that are related to the definiens. For the definiendum, the types of articles, adjectives and nouns used have been considered. The linguistic characteristics of the definiens that have been examined are the articles, adjectives, and relative clauses.

Article in definiendum Fahmi and Bouma (2006) investigated for *is* definitions whether there is a connection between the type of article (definite, indefinite, other) in the definiendum and the class of sentences (definition or non-definition). In their data set, which consists of *is* definition candidates extracted from Wikipedia, the majority of subjects in definition sentences do not contain an article (63%), while in non-definition sentences the use of definite articles is most common (50%). Although our document collection consists of texts that are less structured than Wikipedia documents, there are several similarities between the *is* sentences from Wikipedia and the definition corpus.

Table 4.6 shows the articles used in the definienda for each of the four types of definitions. For the *is* definitions, there are similarities and differences with the figures for the Wikipedia data from Fahmi and Bouma (2006). The proportion of definite articles in non-definitions is in both data sets high. An important difference between the two data sets is the proportion of definite and indefinite articles in the definitions. In the LT4eL definitions, the proportion of indefinite articles is much higher whereas the proportion of definite articles is twice as low. Since Fahmi and Bouma (2006) do not make a more fine-grained cat-

	is		**verb**		**punctuation**		**pronoun**	
	D	ND	D	ND	D	ND	D	ND
article								
- *indefinite*	24.53	8.11	17.31	10.66	7.07	7.87	6.49	16.18
- *definite, 'de'*	3.14	27.36	18.59	27.41	3.03	13.70	6.49	12.83
- *definite, 'het'*	9.43	13.85	10.90	16.75	2.02	6.71	2.6	6.95
no article, noun								
- *common, sg*	10.06	14.86	18.59	11.68	32.32	29.37	12.99	14.97
- *common, pl*	8.81	6.08	14.10	6.09	14.14	22.01	5.19	16.18
- *proper, sg*	43.40	28.38	18.59	26.90	39.39	19.68	2.60	2.27
- *proper, pl*	0	0.68	0	0	0	0.07	0	0
other	0.63	0.68	1.92	0.51	2.02	0.58	3.90	6.69
no definiendum							59.74	23.93

Table 4.6 *The use of articles in definiendum of definitions (D) and non-definitions (ND) (in %). For definienda lacking an article, the type of noun is provided.*

egorization within the 'no article' group, it is not possible to compare the data sets with respect to this. In the LT4eL data, both in definitions and non-definitions the majority of definienda lacking an article contain proper nouns.

The differences observed for the *is* definitions are not seen to the same extent in the *verb*, *punctuation* and *pronoun* definitions. In the *verb* data set, use of definite articles is more common than indefinite articles. Just as for the *is* definitions, the proportion of indefinite articles is considerably higher in the definitions (17% versus 11%). For the *punctuation* patterns, the most notable result is the low proportion of definienda in definitions containing an article, which is as low as 12%. Furthermore, the high proportion of proper nouns in the definitions (38%) is notable. This is similar to the amount of proper nouns used in *is* definitions.

Noun in definiendum Another type of information that seems to be useful for the identification of *is* definitions, is the distinction between proper and common nouns. Fahmi and Bouma (2006) found that proper nouns are more common in the definiendum of definitions whereas common nouns are used more often in non-definitions. Since the MBT

tagger does not provide more detailed information about the type of proper noun (e.g. person, location), it is not possible to make a more fire-grained distinction. The proper and common nouns can be either singular or plural.

	is		**verb**		**punctuation**		**pronoun**	
	D	ND	D	ND	D	ND	D	ND
common noun								
- *singular*	37.11	55.74	53.85	56.35	41.41	50.87	25.97	46.12
- *plural*	8.18	9.46	14.10	12.18	16.16	25.22	7.79	19.39
proper noun								
- *singular*	54.09	33.45	28.85	29.95	40.40	23.25	2.60	3.74
- *plural*	0	0.68	0	0.51	0	0	0	0
other	0.63	0.68	3.21	1.02	2.02	0.66	3.90	6.82
no definiendum							59.74	23.93

Table 4.7 *The use of nouns in definiendum of definitions (D) and non-definitions (ND) (in %)*

Table 4.7 indicates for each of the four definition categories how often the different types of nouns are used in the definiendum. The comparison reveals that the *is* and *punctuation* definitions more often contain proper nouns in the definiendum than the non-definitions of these types. In the *verb* data set, the different types of nouns are equally represented in the two classes. The amount of *pronoun* definitions containing a definiendum in the same sentence is low. When the definitions missing a definiendum are ignored, the differences between definitions and non-definitions are small. The relatively low proportion of proper nouns compared to the other definition types is most striking.

Adjective in definiendum The type of adjective used in the definiendum has not been investigated in previous research. The motivation to include this feature is based on the observation that comparative or superlative adjectives in definienda of definitions typically indicate that a sentence is not a definition. Although none of the definitions from the LT4eL corpus contains a comparative or superlative adjective, this does not mean that they cannot be used in definitions at all. Since the grammars had to be as general as possible to obtain a high recall, this

restriction has not been implemented there.

	is		**verb**		**punctuation**		**pronoun**	
	D	ND	D	ND	D	ND	D	ND
no adjective	93.08	80.07	89.74	86.29	82.83	83.67	29.87	62.43
adjective								
- *positive*	6.29	15.88	8.97	11.17	16.16	14.29	9.09	11.63
- *comparative*	0	0.34	0	0	0	0.66	0	0.27
- *superlative*	0	3.38	0	1.52	0	0.44	0	0.53
adverb	0.63	0.34	1.28	1.02	1.01	0.95	1.3	1.2
no definiendum							59.74	23.93

Table 4.8 *The use of adjectives in definiendum of definitions (D) and non-definitions (ND) (in %)*

Table 4.8 shows that adjectives in the definiendum are most common in *punctuation* definitions (17.2%) whereas they are used less in the *is* definitions (6.9%). The biggest difference between definitions and non-definitions is observed within the *is* data set, where the frequency of adjectives is considerably higher in the non-definitions. As expected, comparative and superlative adjectives have not been used in definitions. However, they are rarely used in non-definitions as well. The positive adjective is most common in both the definitions and the non-definitions. This is the same in the complete LT4eL corpus, where 91.6% of the adjectives is positive. The adverb category contains words tagged as adverbial adjectives, which can be used either as adverb or as adjective (e.g. 'technologisch' (*technologic*) and 'automatisch' (*automatic*)). Most of them should have been tagged as positive adjectives instead of adverbial adjectives.

Article in definiens Fahmi and Bouma (2006) investigated the use of articles at the beginning of the definiens of *is* sentences and discovered that the majority of predicative complements begins with an indefinite article in definitions, while in non-definitions most articles are definite. I compared the definitions and non-definitions of the different data sets to find out whether this difference is observed in these data as well.

Table 4.9 shows that in our *is* data set, just as in the Wikipedia data, the vast majority of articles at the beginning of the definiens is indefi-

	is		**verb**		**punctuation**		**pronoun**	
	D	ND	D	ND	D	ND	D	ND
article								
- *indefinite*	67.92	29.05	12.82	12.69	12.12	2.55	9.09	4.41
- *definite, 'de'*	9.43	17.91	14.10	12.69	3.03	5.10	5.19	14.57
- *definite, 'het'*	6.29	10.81	12.18	8.63	5.05	4.59	2.60	19.52
no article, noun								
- *common, sg*	1.89	17.23	11.54	12.18	17.17	20.63	41.56	23.13
- *common, pl*	8.18	4.39	21.79	14.72	16.16	7.87	25.97	13.37
- *proper, sg*	1.89	6.76	5.77	1.02	0	3.06	3.90	3.88
- *proper, pl*	0	0	0	0	0	0.15	0	0
adverb	3.14	5.74	3.85	12.18	12.12	12.17	1.30	2.67
other	1.26	8.11	17.95	25.89	34.34	43.88	10.39	18.45

Table 4.9 *The use of articles at the beginning of definiens in definitions (D) and non-definitions (ND) (in %)*

nite whereas in the non-definitions this is only the case for a minority of the sentences. An important difference with the Wikipedia data set involves the use of definite articles in the initial noun phrase of the definiens of definitions, which is considerably higher in the data from Fahmi and Bouma (2006). Another difference is the proportion of non-definitions lacking an article, which is much higher in our data.

For the other definition types, the differences with respect to the type of article are smaller than for the *is* sentences. In the *verb* data set, the definiens of definitions more often begins with a definite article than the definiens of non-definitions (26% vs 21%) whereas the proportions of indefinite articles is the same (13%). In definitions, it is more common to start the definiens with a noun than in non-definitions (39% vs 28%). In the remaining sentences, the definiens begins with a different phrase, like an adverb, pronoun or verb. This is more often observed in non-definitions. The fact that *verb* definitions are more diverse makes it more complicated to extract this feature properly. In addition to the pattern 'definiendum - connector - definiens' there are more complex sentences as well. An example of a more complex *verb* definition is:

(13) Met de regelafstand wordt de ruimte tussen de regels
By the line distance is the space between the lines
bedoeld .
meant .
'*Line spacing refers to the space between the lines.*'

The connector phrase is split in two parts ('wordt' (*is*) and 'bedoeld' (*meant*)) and the definiens is contained between these two parts. Due to the greater diversity of patterns, it is less straightforward to detect the article properly. As a consequence, the extraction results contain some errors, which have not been corrected.

Compared to the *is* and *verb* definitions, relatively few definientia in *punctuation* sentences begin with an article. Within this data set, the main difference between definitions and non-definitions concerns the use of indefinite articles, which is observed more frequently in definition sentences (12% versus 3%). In many *punctuation* sentences, the definiens does not begin with a noun phrase. Common word categories observed at the beginning of the definiens are adverbs, pronouns and verbs.

As described in Section 2.3.4, within the *pronoun* patterns a number of different types are distinguished. The article feature is only relevant for the pronoun-is and pronoun-verb sentences, since in the other types the definiens does not begin with a noun phrase. In the pronoun-is and pronoun-verb sentences, the definiens part of definitions generally begins with a noun whereas in the definiens of non-definitions, articles are more common.

Adjective in definiens The next feature that has been examined is the adjective that is used in the first noun phrase of the definiens. Table 4.8 has shown that the adjective in the definiendum is used differently in definitions and non-definitions of the *is* data set. This feature investigates whether there is a difference with respect to the use of adjectives in the initial noun phrase of the definiens as well.

Table 4.10 shows that adjectives in the first noun phrase of the definiens are most common in *is* definitions. For the three other definition types, the frequencies are lower. Whereas for the adjectives in

	is		**verb**		**punctuation**		**pronoun**	
	D	ND	D	ND	D	ND	D	ND
no adjective	77.36	78.72	88.46	85.79	88.89	89.8	88.31	84.49
adjective								
- *positive*	21.38	18.58	10.9	12.69	11.11	9.55	10.39	13.64
- *comparative*	0.63	1.69	0	0	0	0.29	0	0.67
- *superlative*	0.63	0.34	0	1.52	0	0.07	1.3	0.4
adverb	0	0.68	0.64	0	0	0.29	0	0.8

Table 4.10 *The use of adjectives in the definiens in definitions (D) and non-definitions (ND) (in %)*

the definiendum there is a difference between the definitions and non-definitions, this difference is not observed for the adjectives in the definiens. Again, by far the most common category for all types in both definitions and non-definitions are the positive adjectives.

Relative clause Manual inspection of the *is* data set revealed that *is* definitions often contain a relative clause in which a more detailed analysis of the definiendum is provided. Muresan and Klavans (2002) noticed this as well and used this information in their pattern-based extraction approach. In terms of genus-species definitions, the relative clause contains the differentia that are typical for the genus (definiendum) and are needed to distinguish it from other genera within the same species. An example of such a definition is:

(14) Een vette letter is een letter die zwarter wordt afgedrukt dan
A bold letter is a letter that blacker is printed than
de andere letters .
the other letters .
'*A bold letter is a letter that is printed blacker than the other letters.*'

In this example, the genus 'bold letter' belongs to the species 'letters'. It is differentiated from other letters by adding a relative clause which tells that such a letter is printed blacker than the other letters. Based on this observation a feature has been included that considers whether or not the sentence contains a relative clause. Relative clauses

can begin with five types of elements (Coppen et al., 2002), two of which are observed in definitions. More specifically, these are the relative pronouns ('dat', 'die'), and the pronominal adverbs (e.g. 'waarmee', 'waarbij'). A third type of element - that has not been mentioned by (Coppen et al., 2002) - are the words 'om' (*to*) and 'voor' (*voor*). Example 15 shows that the three types – relative pronoun, pronominal adverb, and preposition (including conjunction) – can all be used to convey the same message:

(15) a. PowerPoint is een programma dat gebruikt kan worden
PowerPoint is a program that used can be
voor het maken van presentaties .
for it making of presentations .
'*PowerPoint is a program that can be used for making presentations.*'

b. PowerPoint is een programma waarmee je
PowerPoint is a program with which you
presentaties kunt maken.
presentations can make .
'*PowerPoint is a program that allows you to make presentations.*'

c. PowerPoint is een programma om presentaties mee te
PowerPoint is a program to presentations with to
maken .
make .
'*PowerPoint is a program for making presentations.*'

Table 4.11 shows the differences between definitions and non-definitions for the three types of relative clauses. The first part of the table is about the clauses starting with a relative pronoun (relative clauses), the second part provides information on pronominal adverbs, and the third part indicates how often preposition phrases are used. As can be seen in the last row, the *is* definitions contain most relative clauses (74.2%). The difference between definitions and non-definitions with respect to the proportion of sentences that contain a relative clause is almost 50%. In the *punctuation* and *pronoun* definitions, the definitions

	is		**verb**		**punctuation**		**pronoun**	
	D	ND	D	ND	D	ND	D	ND
relative pronoun								
- 'die' (*who*)	22.01	9.80	11.54	13.20	10.10	3.13	15.58	4.68
- 'dat' (*that*)	18.87	4.39	4.49	3.05	3.03	1.09	11.69	4.14
pron. adverb								
- 'waarmee'	6.92	0.68	0	2.03	1.01	0.07	2.60	0.13
- 'waarbij'	4.40	1.01	0.64	0	0	0.36	3.90	0.67
- 'waarin'	2.52	1.01	1.92	1.52	3.03	0.29	5.19	1.07
- 'waarvan'	0.63	0.34	0.64	0.51	0	0	0	0
preposition								
- 'om' (*to*)	11.32	3.72	5.13	4.06	5.05	0.87	2.60	2.67
- 'voor' (*for*)	7.55	5.41	1.92	5.08	11.11	2.55	6.49	4.28
none	25.79	73.65	73.72	70.56	66.67	91.62	51.95	82.35

Table 4.11 *The use of relative clauses in the definiens of definitions (D) and non-definitions (ND) (in %)*

contain respectively 25% and 30% more relative clauses than the non-definitions. The only data set for which this difference is not observed, is the *verb* data set, in which the non-definitions contain slightly more relative clauses.

Within the *is*, *punctuation* and *pronoun* data sets, relative pronouns are used three times more often in definitions than in non-definitions. The difference is even bigger for the relative clauses starting with a pronominal adverb, where definitions contain four times more relative clauses than non-definitions. For the clauses starting with a preposition, the differences depend on the definition type. The largest differences are observed within the *is* and *punctuation* data sets, in which the definitions contain more relative clauses than the non-definitions. For the *pronoun* sentences, there is only a difference with respect to the use of clauses starting with 'voor', which is observed more in definitions than in non-definitions. The situation is quite different for the *verb* data set, in which the three types of relative clauses are used equally often in definitions and non-definitions.

4.4.3.4 Position properties

Blair-Goldensohn et al. (2004) distinguish two types of position information in their work on the extraction of definitions. The first one is the 'term concentration', that is, the frequency of the term for which a definition needs to be found within a sentence and the nearby context. The other is related to the relative and absolute position of a sentence in a document and is based on the observation that important information is often found towards the top of documents. A specific type of documents are Wikipedia articles, in which the first sentence generally is a definition. Fahmi and Bouma (2006) used this observation and included a binary feature on the sentence position that indicates whether or not a definition candidate is the first sentence of the document. This feature is less useful in our documents, which are not as structured as Wikipedia articles. In addition to being less structured, our texts are often longer and generally contain definitions on several aspects of a topic. For example, a document on XML can include definitions for related terms as well, such as XML Schema, RDF, and HTML. A document from the LT4eL corpus contains on average 13.4 definitions, so the first position feature would not be very informative.

A number of position features have been investigated. These concern the relative and absolute position of sentences within the paragraph, the start position of definition candidates within the sentence and the relative and absolute position of the definiendum compared to other occurrences of the term in the document. None of these features has been investigated in research on definition extraction before.

Position in paragraph The corpus documents consist of paragraphs that are divided into sentences. A paragraph is a 'distinct passage or section of a text, usually composed of several sentences, dealing with a particular point, a short episode in a narrative, a single piece of direct speech, etc. Paragraphs [...] are now usually indicated by a line break, often with an indent at the beginning of the new line.'[1] Each paragraph can thus be considered as a separate block of information. One would expect definitions to appear more at the beginning of such a block than

[1]Source: Oxford English Dictionary (2009).

non-definitions. In order to assess whether this intuition is correct, the length of the paragraph and the position of the sentence within this paragraph are relevant. Three types of position features are therefore included:

- length of the paragraph (LP)
- absolute position of sentence in the paragraph (AP)
- relative position of sentence in the paragraph (RP)

Length paragraph In the section on layout features, it has been shown that the majority of definitions is used in a paragraph. However, since this is not the case for all definitions – they can be used in list items or tables as well – it is necessary to include all paragraph types.

	D	ND	t	Sig
is	4.92	4.56	0.79	0.22
verb	4.85	4.13	1.69	<0.05
punctuation	2.77	3.61	-2.98	<0.01
pronoun	5.25	4.75	1.11	0.13
all	4.52	4.08	2.12	<0.05

Table 4.12 *The length of paragraphs containing definitions (D) and non-definitions (ND) (in sentences)*

Table 4.12 shows that the average length of paragraphs differs for each data set. For the *is*, *verb* and *pronoun* sentences, the paragraphs with a definition are on average longer than the paragraphs containing a non-definition. For the *punctuation* sentences, the situation is different. In the layout features, it has already been shown that *punctuation* sentences are more often observed in list items than the other definition types. Nearly half of the paragraphs containing a *punctuation* definition consists of one sentence. The differences in length between definitions and non-definitions are only significant for the *verb*, *punctuation*, and combined data sets.

Absolute position The position within the paragraph can be measured as an absolute or a relative property. In this feature the abso-

lute position (AP) is considered, by which the position of the sentence within the paragraph is meant.

	D	ND	t	Sig
is	2.02	2.53	-2.20	<0.05
verb	2.07	2.37	-1.25	<0.05
punctuation	1.64	2.41	-5.17	<0.001
pronoun	2.79	2.83	-0.13	0.45
all	2.08	2.54	-4.32	<0.001

Table 4.13 *The average position of definitions (D) and non-definitions (ND) within the paragraph (in sentences)*

Definitions are used earlier within the paragraph than non-definitions (Table 4.13). For three of the four definition types and the combined data set, there is a significant difference between the absolute position of definitions and non-definitions (*is*, *verb*, and *punctuation* definitions). There is no significant difference for the *pronoun* patterns. This can be related to the fact that *pronoun* definitions are often spread over two sentences, of which only the second is addressed by the extraction method.

Relative position In addition to the absolute position of a sentence, a score on the relative position (RP) has been included as well. This score relates the absolute position (AP) to the length of the paragraph (LP). The intuition is that definitions are used relatively early in the paragraph, since they introduce new concepts which can be worked out in more detail in the remainder of the paragraph. The relative position has not been addressed earlier in this way. Other approaches (Blair-Goldensohn et al., 2004; Fahmi and Bouma, 2006) have used the position of a sentence within the document instead of the paragraph. However, since many of our documents discuss different topics and contain more definitions – an average document contains 13.4 definitions – the position within the document is less relevant.

$$\mathrm{RP} = \frac{\mathrm{AP}}{\mathrm{LP}} \tag{4.3}$$

This relative position formula (Formula 4.3) assigns a score between zero and one to each of the definition candidates. It should be noted that the score is highly influenced by the length of the paragraph. In a paragraph of one sentence, the relative position is 1 according to this formula, since it is at the last position of the sentence. However, it is at the first position of the paragraph as well. The relative position formula does not take this problem into account. The combination with the absolute position makes it possible to relate the absolute and relative position.

	D	ND	t	Sig
is	0.48	0.64	-2.82	<0.001
verb	0.53	0.68	-2.28	<0.001
punctuation	0.76	0.77	-0.08	0.41
pronoun	0.61	0.69	-0.98	<0.05
all	0.57	0.72	-4.42	<0.001

Table 4.14 *The average relative position of definitions (D) and non-definitions (ND) within the paragraph (in sentences)*

The relative position (Table 4.14) is significantly lower in definitions than non-definitions within the *is*, *verb*, and *pronoun* data sets, which shows that the definitions are used relatively earlier in the paragraph than the non-definitions. The difference for the *punctuation* sentences is not significant, which means that the relative position is the same for the two classes in this data set. This can be related to the fact that 48.5% of the paragraphs containing a *punctuation* definition consist of only one sentence. For all these sentences, the relative position is 1 (1/1) with this formula.

Position in sentence One of the differences observed between the four definition types, is the place in the sentence where definitions can start and end. Whereas *is* and *verb* definitions tend to span a complete sentence, the rules for *punctuation* definitions are less strict for this feature. On the basis of this observation, it has been investigated whether this type of position information can be used to distinguish definitions from non-definitions.

A more pragmatic reason to investigate this property is related to the pattern-based approach. As explained in Section 3.5.1 (rule AG8 of the grammars), the beginning of a sentence can be described with two rules. The first rule matches the first token element of a sentence, whereas the second matches all capitalized words as possible initial elements. Since capitalized words can have other functions as well – the most common one being a proper noun – this second rule may result in the extraction of non-definitions. The position is indicated by the number of the first token of the definition within the <s> element. So, if a position begins at the start of an <s> element, the score is 1.

	D	ND	t	Sig
is	1.24	2.44	-3.70	<0.001
verb	1.29	3.06	-3.97	<0.001
punctuation	5.34	9.71	-5.52	<0.001
pronoun	3.29	8.04	-6.72	<0.001
all	2.4	7.91	-19.34	<0.001

Table 4.15 *The average beginning position of definitions (D) and non-definitions (ND) within the sentences (in tokens)*

Definitions begin significantly earlier in the sentence than non-definitions within all data sets. The majority of *is* and *verb* definitions start at the beginning of the sentence (97% and 94%). The situation is different for the *punctuation* and *pronoun* definitions. Only 48% of the *punctuation* definitions starts at the beginning of a sentence whereas for the *pronoun* definitions this is 74%.

Position of definiendum The function of a definition is to explain the meaning of a term. One would therefore expect that terms are defined one of the first times they are used. It sounds odd to define a term after it has been used a lot of times already. Although it is possible that it has been used two or three times before the definition (e.g. in title of document, table of contents or heading), intuitively one would expect it to occur more frequently after it has been defined. Based on this intuition, five features have been included. The first three are related to the number of times the term defined in the definiendum has been used in the

complete document and provide information on the absolute position. Two other features assess the relative position of the definiendum. The position of the definiendum within the document has not been examined before.

Frequency and absolute position The first two measures are the absolute number of occurrences of the term before and after it is used in the definition candidate. For three of the types, the average number of occurrences before is significantly lower for all types except the *is* definitions. The number of occurrences of the term after it has been defined seems to be a less good predictor and is only significantly lower for the *is* definitions.

	D	ND	t	Sig	D	ND	t	Sig
		Before				After		
is	4.69	4.85	-0.1	0.46	22.38	5.76	3.43	<0.001
verb	2.65	5.73	-2.28	<0.05	7.04	7.87	-0.35	0.36
punctuation	2.92	10.91	-5.17	<0.001	5.51	10.02	-2.16	<0.05
pronoun	0.52	6.41	-8.08	<0.001	4.42	7.53	-3.56	0.12
all	3.03	8.54	-6.30	<0.001	11.29	8.66	5.85	0.12
		All						
is	27.07	10.61	2.96	<0.01				
verb	9.69	13.59	-1.16	0.12				
punctuation	8.42	20.93	-3.85	<0.001				
pronoun	4.94	13.93	-3.51	<0.001				
all	14.32	17.2	-1.24	0.11				

Table 4.16 *Frequency of use of the defined terms from definitions (D) and non-definitions (ND) in the complete document*

The figures from Table 4.16 are in line with the intuitive expectation with respect to the distribution of the definiendum in definitions and non-definitions. The upper left part of the table refers to frequency before the definition candidate is used. The upper right part focuses on the frequency after the term has been defined and the other part gives the total number of times the definiendum has been used in the document.

The number of the definiendum before it is used in the definition candidate is similar in definitions and non-definitions from the *is* data set. For the other terms, the frequencies are significantly lower for the

definitions than the non-defitions. A possible explanation for this could be that the definienda of the *is* definitions are so important that they are used in titles, headers and the table of contents more often than the definienda of the other definition types. In the other data sets, the difference in this respect is especially big for the *punctuation* and *pronoun* patterns.

The difference between definitions and non-definitions is less clear when it comes to the number of times the definiendum is used after the definition candidate. Again, the *is* definitions are an exception, since for this type the terms defined in the definitions are used considerably later than the terms from the non-definitions. For the *punctuation* data set, there is a significant difference as well. However, here the definienda from the non-definitions are observed more often after the definition candidate.

The total frequency of the terms depends on the definition type as well. Again, there is a distinction between the behaviour of the *is* definitions on the one hand and the three other definition types on the other hand. In the *is* data set, the frequency of the definienda is significantly higher in the definitions than in the non-definitions. For the *verb* definitions, there is no significant difference, although the average number is higher here for the non-definitions. In the *punctuation* and *pronoun* data sets, the terms from the non-definitions are used significantly more. There is no significant difference for the complete data set, which can be explained by the differences observed for each type.

Relative position The difference between the frequencies before and after the definiendum are considerably bigger for the definitions than for the non-definitions (Table 4.16). Based on this observation, I identified two formulas to express the relative position of the definiendum within the document. This information has not been used before and the formulas proposed are both new.

The first formula looks at the relative position of the definiendum within the document. It is similar to the formula for the relative position of the sentence within the paragraph (Formula 4.3). The absolute position has been replaced by the number of times the term has been used earlier in the document plus one for the occurrence of the term in

the definition. The length of the paragraph in this equation is the total frequency of the term. Formula 4.4 shows how the relative position has been measured:

$$\frac{1 + \text{frequency before}}{\text{total frequency}} \tag{4.4}$$

In Formula 4.5, the balance between the amount of times the term is used before and after the definition is considered. Compared to the relative position formula, an advantage of the balanced formula is that it is less sensible to the amount of times the term is used in the document, which is useful since a lot of the terms are not frequently used. It simply shows whether a term is relatively more to the beginning or more to the end of a document. A disadvantage of both formulas is that they do not reflect the frequency of the term, and therefore it is important to include the absolute frequency information as well. The balanced formula returns a score between -1 (definition contains last occurrence definiendum) and 1 (definition contains first occurrence definiendum):

$$\frac{\text{frequency after} - \text{frequency before}}{\text{frequency after} + \text{frequency before}} \tag{4.5}$$

	Relative position				Balance			
	D	ND	t	Sig	D	ND	t	Sig
is	0.59	0.78	-5.68	<0.001	0.2	0.03	2.76	<0.01
verb	0.67	0.77	-2.71	<0.01	0.17	-0.01	14.72	<0.01
punctuation	0.79	0.78	0.20	0.30	0.13	0.01	2.57	<0.01
pronoun	0.21	0.54	-7.93	<0.001	0.19	0.04	18.88	<0.01
all	0.6	0.71	-5.94	<0.001	0.17	0.02	45.05	<0.001

Table 4.17 *Relative position and balanced position of definiendum in definitions (D) and non-definitions (ND) within the complete document*

Table 4.17 reveals that in most data sets definienda are used significantly earlier in the documents in definitions than in non-definitions. In the *punctuation* definitions, the relative position is the same for the two classes. The balanced score shows the same difference. In the non-definitions, the terms are on average in the middle (scores close to 0),

whereas for the definitions, they are used relatively more towards the beginning of the document (scores between 0 and 1).

For both formulas, the high amount of unique terms largely influences the scores. The relative scores are closer to 1 than expected, since the relative score is 1 for terms that occur only in the definition. The balanced scores do not deviate much from 0 for the same reason.

4.4.3.5 Layout properties

The layout properties consider the structure of the document and the way in which the layout of definitions is different in definitions and non-definitions. The first property is based on the idea that learning objects are structured and divided into different parts (e.g. chapters, sections, subsections), in which each block of text has its own layout or style. In this way it is possible to distinguish, for example, headers from list items and titles from paragraphs. Intuitively, one would expect definitions to be used primarily within the paragraphs. This information is used in the first layout feature. To the best of my knowledge, no other research has included this kind of structural information as a feature for definition classification. A second type of layout information involves the investigation of different font styles. I assume that definitions are more often formatted in a special way (e.g. bold, italics) than non-definitions. This assumption has not been tested and implemented before as a machine learning feature.

Type of paragraph As described in Section 3.4.1, layout information has been preserved in the XML documents and has been stored in the *rend*-attribute of tokens, sentences and paragraphs. At the paragraph level, this attribute specifies the type of paragraph. The fifteen different types are based on the standard HTML notation for structuring texts. The most common paragraph type is the "`p`"-type, which refers to standard text paragraphs. Seven types are related to headings ("`h1`" to "`h6`") and titles ("`title`"). Then there is the encoding for list items ("`li`"), for information from table cells ("`th`" for header cells and "`td`" for data cells), and for definition lists ("`dt`" for terms and "`dd`" for definitions). The code "`div`" is used to define a generic block-level container. The

"pre" code indicates verbatim text and is often used to format computer code or other text in which it is important to preserve the whitespace.

Intuitively, one would expect definitions to appear most often in 'real' paragraphs, denoted by the value p. For the *is*, *verb* and *pronoun* definitions, this indeed is the case (86% - 91%). However, there is not much difference between definitions and non-definitions with respect to the type of paragraph. Within the *punctuation* data set, there is a clear distinction between definitions and non-definitions. Only 55% of the definitions are used in <p>-paragraphs whereas this is 73% for the non-definitions. It is interesting to note that *punctuation* definitions are quite common in list items (26%). Not only the structure of the sentences is different from the other definition types, but also the way in which the definitions are used in the text.

Layout of definiendum Since definitions contain important information, one would expect special layout features (e.g. bold, italics, underlined) to occur more often in definitions than in non-definitions. Information on the layout of the original documents is stored at the word, sentence and paragraph level in the XML files. For this feature, the layout of the nouns in the definiendum is investigated as this is the key term of the definition.

	is		**verb**		**punctuation**		**pronoun**	
	D	ND	D	ND	D	ND	D	ND
layout	18.87	7.77	17.95	8.63	28.28	14.07	9.09	5.75
- *b*	3.77	1.69	3.85	0	10.10	5.25	1.30	1.34
- *i*	10.06	3.38	9.62	5.08	11.11	6.49	6.49	3.88
- *u*	1.89	0.68	0.64	2.54	3.03	1.09	0	0
- *dfn*	1.89	0	0	0.51	0	0	0	0
- *sup*	1.26	1.01	2.56	0	3.03	1.02	1.30	0.53
- *em*	0	0.68	0.64	0	1.01	0	0	0
- *code*	0	0.34	0.64	0.51	0	0.22	0	0
no layout	81.13	92.23	82.05	91.37	71.72	85.93	90.91	94.25

Table 4.18 *The use of layout elements in the definiendum of definitions (D) and non-definitions (ND) (in %)*

Table 4.18 compares the layout information used in the definien-

dum of definitions and non-definitions for the four types of definitions. For each of these types, the definienda of definitions more often contain a specific type of layout than the definienda of non-definitions. The proportion of layout information is about twice as high in definitions than in non-definitions, regardless of the definition type. Since some of the properties are rarely used, all layout features have been combined into one group. This makes it possible to test whether there is a significant difference between the two groups, which is indeed the case: layout is used significantly more in definitions than in non-definitions for three types (*is* definitions ($\chi^2(1) = 11.32, p < 0.001$), *verb* definitions ($\chi^2(1) = 5.99, p < 0.025$) and *punctuation* definitions ($\chi^2(1) = 13.52, p < 0.001$)). For the *pronoun* definitions, the amount of definienda is not high enough to draw statistical conclusions, because not all sentences contain a definiendum part. Two layout features have been included. The first contains information on the separate layout types (such as b, i, u) and the second is a binary feature which indicates whether or not a specific layout has been used.

4.4.3.6 Keyword properties

Definitions provide the reader of a text with an explanation of a term. Intuitively, one would expect more definitions for words or phrases that are important within a text. To indicate the importance of words within a document, it is possible to use keyword scores. A keyword score determines how relevant a words is in a specific text by comparing it to another a corpus of documents.

The keyword extractor that has been developed within the LT4eL project (Lemnitzer and Monachesi, 2008) has been used. This tool assigns a score to each word and phrase in the text that indicates how important the term is within the document. The phrases are restricted on the basis of the lexico-syntactic patterns that keyphrase can have. For example, a combination of an adjective and a noun can be a keyphrase whereas the combination of an article and a noun is not possible.

From the different available keyword extractors, the LT4eL extractor has been selected for three reasons. First, it can deal with Dutch texts, whereas many keyword extractors are available only for English (cf. Sclano and Velardi (2007), Frank et al. (1999), Witten et al. (1999)).

A second important aspect of the LT4eL keyword extractor is that it can extract keyphrases in addition to keywords. This is relevant, since the definiendum often consists of more than one word. The third – more pragmatic – reason to use the LT4eL keyword extractor is that this tool has been designed to work with the LT4eLAna document format.

The keyword scores can be measured by the tool in three different ways. For each of the three measures, it has been investigated whether there is a distinction between the definitions and non-definitions. The scores have been calculated on the basis of the LT4eL corpus.

RIDF keyword score The RIDF score makes use of the Poisson distribution or a mixture of Poisson distributions (Church and Gale, 1995a). While the distribution of function words like *of*, *the*, and *it* and other common words is close to the expected distribution under the Poisson distribution, good keyword candidates deviate significantly from this expectation. The more a word deviates from Poisson, the more useful it is for discriminating documents and thus the more appropriate it is as a keyword (Church and Gale, 1995b). The residual inverse document frequency ($RIDF$) is based on this observation and is calculated by taking the difference between the logarithm of the actual inverse document frequency ('observed IDF') and the inverse document frequency predicted under the Poisson model ('predicted IDF'):

$$\begin{aligned} RIDF &= IDF_o - IDF_p \\ &= \log_2 \frac{N}{df_t} - (-\log_2(1 - e^{-p})) \\ &= \log_2 \frac{N}{df_t} + \log_2(1 - e^{-p}) \\ &= \log_2 \frac{N(1 - e^{-p})}{df_t} \end{aligned} \tag{4.6}$$

with p being $\frac{CF}{N}$ and CF the frequency of the word in the complete corpus. The RIDF score is thus based on the frequency of a word in the complete corpus and the number of documents in which it occurs. The RIDF score can be used for comparison across documents, since the frequency of the term within a document is not taken into account.

	D	ND	t	Sig
is	2.13	1.6	4.72	<0.001
verb	1.81	1.7	0.92	0.22
punctuation	1.05	1.43	-3.29	<0.05
pronoun	2.05	1.52	4.27	<0.05
all	1.79	1.49	5.25	<0.001

Table 4.19 *Average RIDF scores for the definiendum in definitions (D) and non-definitions (ND)*

The expectation about the keyword scores was that the scores would be on average higher for definienda used in definitions than in non-definitions. Table 4.19 shows that this idea is only correct for the *is* and *pronoun* definitions, where especially the difference within the *is* definitions is big. For the *verb* definitions, there is no significant difference and for the *punctuation* definitions the scores are even higher in the non-definitions. For the combined data set, the average keyword score is significantly higher for the definitions than for the non-definitions.

TFIDF keyword score The TF-IDF score is composed of two parts: the term frequency (TF) and the inverse document frequency (IDF). The term frequency indicates how often a term is used within a specific document. The inverse document frequency is an indication of a term as a document discriminator within a collection of documents. It is computed by taking the inverse function of the document frequency of the term. The document frequency is the number of documents that contain this term. The LT4eL keyword extractor computes IDF using the following formula:

$$IDF = \log_2 \frac{N}{DF_t} \tag{4.7}$$

where N is the number of documents and DF_t the document frequency of the term. In TF-IDF, the IDF part is multiplied with the term frequency:

$$TFIDF = TF \times IDF \tag{4.8}$$

Keyword scores are partly based on the frequency of a term within a document. This is a good way of scoring keywords as long as it is not necessary to compare keyword scores across documents, which is the case in the definition classification situation. The length of documents is very different within the corpus. As a consequence, the range of TF-IDF scores varies for each document. For example, a 50,000 word document on XML probably has a higher frequency of the term XML than a 1,000 word document on the same topic and as a consequence the TF-IDF score will be higher for the longer document. However, this does not automatically imply that the word is more important in the document with the higher score. Different techniques have been proposed to make the scores comparable across documents (cf. Salton and Buckley (1988), Singhal et al. (1995), Singhal et al. (1996)).Although the score may not be very useful on its own, it can be relevant in combination with other features, like the feature on the frequency of the definiendum discussed in the section on position information. Therefore, this score has been included as a feature.

	D	ND	t	Sig
is	104.91	58.01	2.88	<0.01
verb	48.54	61.34	-1.19	0.16
punctuation	24.7	54.44	-4.93	<0.001
pronoun	65.29	54.37	0.73	0.34
all	66.14	55.33	1.68	0.09

Table 4.20 *Average TFIDF scores for the definiendum in definitions (D) and non-definitions (ND)*

Only for the *is* and *punctuation* definitions there is a significant difference between definitions and non-definitions with respect to the TF-IDF score (Table 4.20). The results for these types are the same as for the RIDF score. For the *verb*, *pronoun*, and combined data sets, the difference between the two classes is not significant.

ADRIDF keyword score The Adjusted Residual IDF (ADRIDF) score is an adapted version of the RIDF score, the only difference being that in this score the term frequency within the document has been

taken into account:

$$ADRIDF = RIDF\sqrt{TF} \tag{4.9}$$

where TF is the frequency of the term within the document. For this metric, the same restrictions apply as for the TFIDF score, since the document frequency is considered here as well. Just as for the TFIDF score, I expected that it might be valuable in combination with other features and therefore it has been included as well.

	D	ND	t	Sig
is	11.16	7.04	3.49	<0.01
verb	6.86	7.45	-0.63	0.30
punctuation	3.48	6.33	-4.3	<0.001
pronoun	8.79	6.88	1.45	0.22
all	7.92	6.63	2.56	<0.05

Table 4.21 *Average ADRIDF scores for the definiendum in definitions (D) and non-definitions (ND)*

The ADRIDF scores (Table 4.21) are similar to the TFIDF scores from Table 4.20. The same significant differences are observed for the *is* and *punctuation* data sets and no significant differences are observed for the *verb* and *pronoun* data sets. A difference with the TFIDF scores, is that with the ADRIDF score, the average keyword score within the combined data set is significantly higher for the definitions than for the non-definitions.

Summarizing, the four definition types show a different behaviour with respect to the keyword scores of the definienda in definitions and non-definitions. The *is* definitions and non-definitions are most different with respect to the keyword scores. The definienda from the definitions are for this type more important within the documents than the definienda from the non-definitions. For the *pronoun* data set, the same difference is observed, although it is much smaller. The two classes from *verb* data set do not differ significantly with respect to the keywordiness. In the last data set, the *punctuation* definitions, the average keyword score is higher for the non-definitions than for the definitions.

4.4.4 ALGORITHM SELECTION

4.4.4.1 Selecting a classifier

The previous section revealed that there are a number of differences between definitions and non-definitions. On the basis of these differences, classifiers are trained that are able to distinguish definitions from non-definitions. To find an appropriate classifier, it has been investigated which classifiers are used in other research on definition extraction. On the basis of our data sets and task, the most appropriate one has been selected. The same classifier has been used for all data sets and feature settings.

As described in Section 4.2, the distinction between supervised, unsupervised and semi-supervised classifiers is relevant when a classifier needs to be selected. In order to be able to evaluate the performance of the pattern-based approach presented in Chapter 3, a number of definitions had to be selected and annotated manually. The instances from the machine learning data sets can be labeled automatically on the basis of these annotations. Since the data are labeled, a supervised classifier can be used. Previous machine learning experiments for the classification of definitions have used supervised classifiers as well. Fahmi and Bouma (2006) used three types of supervised classifiers in their experiments on the classification of *is* definitions. These are the Maximum Entropy classifier, a naive Bayes (NB) classifier and Support Vector Machines (SVM). Miliaraki and Androutsopoulos (2004) and Androutsopoulos and Galanis (2005) used an SVM classifier. Kobyliński and Przepiórkowski (2008) report on experiments with the Balanced Random Forest (BRF) classifier. In this classifier, the Random Forest classifier is combined with a sampling technique to deal with imbalanced data sets within the area of definition extraction. The BRF classifier outperformed a number of common classifiers, including the naive Bayes classifier (Degórski et al., 2008).

Since some of our data sets are imbalanced, I investigated whether the bagging procedure from Kobyliński and Przepiórkowski (2008) is useful on these data as well. To this end, an experiment has been carried out in which common classifiers are compared against the balanced versions of the same classifiers. The data set that has been used

classifier	R	P	F	AUC
common classifiers				
NB	0.39	0.37	0.38	0.77
SVM	0	0	0	0.5
RF	0.32	0.36	0.34	0.77
balanced classifiers				
BNB	0.56	0.32	0.41	0.77
BSVM	0.56	0.35	0.43	0.76
BRF	0.60	0.35	0.44	0.81

Table 4.22 *Balanced bagging experiments on the aggregated data set using the PoS bigram setting*

is the aggregated data set, which consists of a combination of the four individual data sets. The aggregated data set contains a minority of definitions (16%) and a majority of non-definitions (84%). Table 4.22 shows that the bagging procedure improves the results for each of the classifiers. The recall scores obtained with the classifiers in which no bagging has been used is considerably lower than the scores obtained with the balanced classifiers. The Balanced Random Forest classifier has been used for my experiments and is discussed in more detail in the remainder of this section.

4.4.4.2 The Balanced Random Forest classifier

The Balanced Random Forest classifier is an adapted version of the Random Forest classifier. The Random Forest classifier is a decision tree algorithm in which prediction is not based on one tree, but on an ensemble of trees (Breiman, 2001). This 'forest' of trees is created by taking bootstrap samples D_i of the training data and using random feature selection for tree induction. A bootstrap sample is a sample in which the sampling instances have been selected from a data set D with replacement. This makes it possible that some instances will be contained more than once in each D_i. Predictions are made on the basis of a majority vote, that is, the class predicted by most of the trees is selected. The idea behind Random Forest is used in other classifiers as well and is called *bagging* (**b**ootstrap **agg**regat**ing**).

A disadvantage of the Random Forest approach is that there is a significant probability that a bootstrap sample contains few or even none of the minority class when data sets are imbalanced, as is the cases in the definition classification context. As a consequence, the resulting tree will perform poor at predicting the minority class. A naive way to fix this problem would be to use a stratified bootstrap, that is, taking a sample with replacement from within each class. The problem with this approach is that a large part of the information from the majority class is lost and not used in classification. This is an example of a down-sampling technique. Other sampling techniques that can be used are over-sampling the minority class, or a combination of both down-sampling and over-sampling (Kubat et al., 1998). A completely different approach to tackle the imbalanced data problem is based on cost sensitive learning. In this approach a high cost is assigned to misclassification of the minority class, and at the same time the overall cost is kept as low as possible (Chen et al., 2004).

The Balanced Random Forest method combines a down-sampling technique with the ensemble idea: it down-samples the majority class and grows each tree on a more balanced data set. It is thus a modification of the Random Forest method designed specifically to deal with imbalanced data sets. In this method an adapted version of the bagging procedure is used, the difference being that trees are induced from *balanced* down-sampled data, which differs from normal down-sampling procedure. A majority vote is taken for prediction. The procedure of the Balanced Random Forest (BRF) algorithm is described by Chen et al. (2004):

1. For each iteration in random forest, draw a bootstrap sample from the minority class. Randomly draw the same number of cases, with replacement, from the majority class.

2. Induce a classification tree from the data to maximum size, without pruning. The tree is induced with the CART (Classification and Regression Trees) algorithm (Breiman et al., 1984), with the following modification: at each node, instead of searching through all variables for the optimal split, only search through a set of m randomly selected variables. Breiman (2001) experimentend with

m = 1 and a higher value of m and concluded that the procedure is not very sensitive to the value of m.

3. Repeat the two steps above for the number of times desired. Aggregate the predictions of the ensemble and make the final prediction.

4.4.5 TRAINING AND TEST SET

The classifiers are trained and evaluated using n-fold cross validation with n being 10. The 10-fold cross validation has been chosen since some of the data sets are relatively small and in this way it is possible to train in each fold on 90% of the original data set. The classifiers have not been evaluated on a different data set.

4.4.6 EVALUATION METRICS

In section 4.2.6, the AUC score has been introduced. Since this score is less sensitive to imbalanced data than the accuracy, it has been used to evaluate the performance of the classifiers. The AUC provides only information on the overall accuracy of the classifier and not on the performance on the individual classes. For this reason, the precision (P_{def}), recall (P_{def}) and F-score (P_{def}) of the definition class are reported as well. The recall and F-score are reported in addition to the AUC and precision to provide a complete view on the performance of a classifier.

4.4.7 PARAMETER TUNING

The parameter tuning component is one of the last components from the machine learning diagram. It is concerned with finding the best settings for the parameters. A distinction is made between parameters related to the features and classifier parameters. The only feature for which different parameter settings had to be compared, are the PoS and MSI bigrams. The other features are either numeric (e.g. the position scores) or nominal (e.g. the linguistic properties). In both cases, the possible values do not need to be reduced. The optimal number of bigrams to be included depends on the number of bigrams available in the data set and the classifier used. To investigate which values give the

best results, the Balanced Random Forest classifier has been extensively tested with different numbers of bigrams. The parameters of the Balanced Random Forest classifier have been tuned as well. The number of iterations to be used in the bagging procedure is the most important parameter that needs to be investigated in this respect.

This section describes the experiments that have been carried out with respect to the tuning of the number of bigrams to be considered (Section 4.4.7.1) and the parameter tuning related to the Balanced Random Forest classifier (Section 4.4.7.2).

4.4.7.1 Bigrams

The first thing that has been investigated is the optimal number of bigrams. Two types of bigrams have been considered: PoS tag bigrams (PoS) and bigrams containing morpho-syntactic information in addition to the PoS tags (MSI). Logically, the number of PoS unigrams is much lower than the number of MSI unigrams. For the first (PoS), there are only 12 classes, whereas for the second (MSI) 175 classes are distinguished within the complete data set containing all definitions and non-definitions extracted with the pattern-based approach. If each sequence of word types were possible, the maximum number of bigrams could be calculated by squaring the unigrams, which would result in 144 PoS tag bigrams (12^2) and 30,625 MSI bigrams (175^2). The two aspects that have been considered in the search for the optimal number of bigrams are the size of the data set and the balance of the classes within the data set.

The size of the data set correlates with the number of different bigrams that are used. Since a large part of the bigrams - and even unigrams - is rarely used, in smaller data sets many of those rare bigrams and unigrams are missing. As a consequence, in the smaller data sets considerably less than the 175 possible MSI classes have been used. For example, the data set for the *is* definitions contains only 119 classes. The largest data set is the set in which all four definition types are included. This data set contains 131 PoS bigrams and 3,187 MSI bigrams. The smallest data set, the *is* data set, contains only 112 PoS bigrams and 1200 MSI bigrams are used. A second relevant issue to be considered with respect to the bigrams is the distinction between balanced

and imbalanced data sets. In the imbalanced data sets, the minority class contains considerably less bigram types than the majority class. For this reason, the bigram values in the experiments have been set for each class, selecting from both classes an equal number of bigrams.

Based on these two aspects, the values for the PoS tag bigrams have been chosen between 0 and 120 whereas for the MSI bigrams the values differ between 10 and 1000. Experiments to find optimal settings have been carried out on all data sets for the Balanced Random Forest classifier, since this is the classifier that will be used in the experiments. The experiments reveal that for each of the data sets the AUC scores stabilize around the 40 bigrams. The largest improvement is observed between including 10 and 20 bigrams. The optimal number of bigrams when including the morpho-syntactic information as well differed slightly for each data set. The results stabilize around the 400 bigrams and therefore this value has been used in the experiments. The results of the experimens can be found in Appendix F.

4.4.7.2 Balanced Random Forest

As described in section 4.4.4, the Balanced Random Forest classifier has been used for the experiments. For this classifier, two parameters need to be tuned. These are the number of iterations to be used in the bagging procedure and the number of attributes to be considered at each node.

Balanced bagging settings In balanced bagging, the bagging procedure is repeated a specified number of times (I). Breiman (1996) experimented with different values of I to find the optimal value and found out that after 25 replicates no further improvement was obtained. Experiments on our data set have been carried out with the value of I varying between 0 (no bagging) and 100 for the Random Forest classifier using 40 PoS bigrams or 400 MSI bigrams. The experiments revealed that both for the PoS and the MSI bigrams the biggest improvement is obtained from 1 replicate (no bagging) to including 10 replicates. Higher values result in slightly improved results, but after 30 bootstrap replicates (or less for some of the data sets) the AUC stabi-

lizes and no significant improvement is observed anymore. Therefore, this value ($I = 30$) has been used in all balanced bagging experiments. Appendix F contains the results of the experiments.

Number of attributes The number of attributes M differs for each data set. In the Balanced Random Forest classifier, the parameter m indicates the number of randomly selected attributes m to be considered at each node. Experiments by Breiman (2001) with $m = 1$ and higher values of m showed that the procedure is not very sensitive to the value of m. Since the value of m thus does not influence the results very much, in the definition classification experiments the same value of m has been used for all experiments, namely $m = 20$.

4.5 CONCLUSIONS

The process of applying machine learning consists of several steps, which are closely related to each other. This chapter presented a general overview of the different components. More specifically, it describes the data identification, data pre-processing, feature selection, algorithm selection, creating training and test set, deciding on evaluation metrics, and parameter tuning. After this general overview, a description is provided of the choices that have been made for the set-up of the machine learning experiments.

In this thesis, I focused on the feature selection component. More specifically, I investigated which types of information that can be extracted from the documents itself can be used for the classification of definitions and non-definitions. Six types of information have been considered. The first type of information are the linguistic bigrams. The other five types are related to connector information, linguistic information of definiendum and definiens, position information, layout information, and keyword information. The comparison of definitions and non-definitions with respect to these features revealed that the definition type plays an important role. The largest differences within the data set are observed in the *is* data set while the differences are smallest in the *verb* data set.

In the algorithm selection component, the Balanced Random Forest classifier has been selected, since this classifier can deal better with imbalanced data sets than other classifiers that have been used for definition extraction (like naive Bayes and SVM). In the parameter tuning experiments, the optimal settings for the experiments has been determined. The last step in the machine learning process is the training of the classifiers. This part is discussed in the next chapter, which presents the results obtained with the different classifiers.

Part of the inhumanity of the computer is that, once it is competently programmed and working smoothly, it is completely honest.

Isaac Asimov (1920-1992)

5 Machine learning results

5.1 INTRODUCTION

Chapter 4 described how the different machine learning components distinguished by (Kotsiantis, 2007) have been addressed in my experiments. The emphasis of the previous chapter was on the selection of appropriate features, which have been used to train classifiers. The results obtained with the different classifiers are discussed in this chapter. The figures presented indicate the final performance after applying the sequential combination of the pattern-based approach and machine learning. An inevitable consequence of using such a sequential combination is that the final recall scores are lower than the recall scores obtained with the pattern-based approach alone. The decrease of this score is the price that has to be paid for the improvement of the precision.

The first part of this chapter describes the relevance of the individual settings (Section 5.2). For each of the characteristics discussed in Section 4.4.3, it has been investigated how they contribute to the classification of definitions. The use of more than one type of information has been examined in Section 5.3, where different combinations of settings are compared to find out which one performs best. Section 5.4 presents the performance of classifiers in which PoS and MSI bigrams have been included in addition to the other characteristics.

5.2 INDIVIDUAL SETTINGS

Section 4.4.3 introduced six groups of text characteristics that can be relevant for the classification of definitions. More specifically, I examined whether the use of linguistic bigrams, connector properties, linguistic properties of definiens and definiendum, position properties, layout properties, and keyword properties is different in definitions and non-definitions. The first part of this section focuses on the sub settings within these six groups and investigates how relevant they are. It then proceeds with a description of the performance of the complete settings to determine for each definition type which kind of information is most relevant for the classification of definitions. The section ends by providing an overall type-based ranking of the relevance of the different settings.

5.2.1 SUB SETTINGS WITHIN INDIVIDUAL SETTINGS

5.2.1.1 *n*-gram properties

The first feature types are the Part of Speech (PoS) bigrams and the morpho-syntactic (MSI) bigrams. Since the parameter tuning experiments from Section 4.4.7 revealed that including more than 40 PoS bigrams or 400 MSI bigrams does not improve the performance, these values have been used in all experiments in which bigrams are included.

Table 5.1 shows the results for the two types of bigrams together with the results of the pattern-based approach. The precision scores obtained with the PoS or MSI bigrams are considerably higher than the precision scores from the grammars alone. For the *punctuation* and *pronoun* types, the precision with the bigrams is (almost) two times higher than the grammar precision. The situation is worse for the *verb* definitions, where the increase of the precision does not compensate the decrease of the recall. As a consequence, the F-scores drop for this type when bigrams are used.

In some of the data sets (e.g. *all* or *pronoun*), there is only a minor difference between the results obtained with only the PoS bigrams or the combination of PoS and MSI bigrams. Although the MSI bigrams

	AUC	R	P	F	AUC	R	P	F
		is				**verb**		
grammar		0.86	0.35	0.50		0.78	0.44	0.56
PoS bigrams	0.70	0.57	0.49	0.52	0.64	0.48	0.52	0.50
MSI bigrams	0.80	0.59	0.57	0.58	0.66	0.46	0.57	0.51
		punctuation				**pronoun**		
grammar		0.88	0.07	0.12		0.75	0.09	0.16
PoS bigrams	0.77	0.63	0.15	0.24	0.69	0.42	0.16	0.24
MSI bigrams	0.79	0.56	0.18	0.28	0.66	0.34	0.14	0.20
		all						
grammar		0.82	0.16	0.26				
PoS bigrams	0.81	0.60	0.35	0.44				
MSI bigrams	0.82	0.56	0.38	0.45				

Table 5.1 *Results for the bigram features*

provide more detailed information on the categories of words than the PoS bigrams, this does not always lead to a better performance in these experiments.

5.2.1.2 Connector properties

Experiments have been conducted with combinations of three features related to the connector. The first feature examines the connector phrases that have been used in definitions and non-definitions. Section 4.4.3.2 showed that the definitions can be divided in two groups when it comes to the variety of such phrases. One group contains the *is* and *punctuation* definitions, in which respectively one or two connector phrases can be used, while the other group consists of the *verb* and *pronoun* phrases, in which the diversity is much higher. The other text characteristics related to the connector are two types of linguistic context information, namely the Part of Speech tags of the words directly to the left and right of the connector phrase (second feature), and the morpho-syntactic information of the words to the left and right.

Connector phrase Table 5.2 shows that for *is* definitions the results of the classifier are exactly the same as the results obtained with the grammar and that the AUC score is quite low. This can be explained by the

	AUC	R	P	F	AUC	R	P	F
		is				**verb**		
Baselines								
grammar		0.86	0.35	0.50		0.78	0.44	0.56
PoS bigrams	0.70	0.57	0.49	0.52	0.64	0.48	0.52	0.50
MSI bigrams	0.80	0.59	0.57	0.58	0.66	0.46	0.57	0.51
Connector								
connector phrase	0.50	0.86	0.35	0.50	0.70	0.51	0.59	0.55
+ PoS left and right	0.61	0.73	0.44	0.54	0.69	0.51	0.57	0.54
+ MSI left and right	0.71	0.66	0.50	0.57	0.72	0.52	0.60	0.56
all	0.71	0.66	0.50	0.57	0.72	0.54	0.61	0.57
		punctuation				**pronoun**		
Baselines								
grammar		0.88	0.07	0.12		0.75	0.09	0.16
PoS bigrams	0.77	0.63	0.15	0.24	0.69	0.42	0.16	0.24
MSI bigrams	0.79	0.56	0.18	0.28	0.66	0.34	0.14	0.20
Connector								
connector phrase	0.60	0.60	0.10	0.17	0.76	0.58	0.23	0.33
+ PoS left and right	0.70	0.61	0.12	0.20	0.84	0.62	0.23	0.34
+ MSI left and right	0.74	0.60	0.14	0.22	0.82	0.60	0.25	0.35
all	0.74	0.59	0.13	0.22	0.83	0.60	0.24	0.35
		all						
Baselines								
grammar		0.82	0.16	0.26				
PoS bigrams	0.81	0.60	0.35	0.44				
MSI bigrams	0.82	0.56	0.38	0.45				
Connector								
connector phrase	0.79	0.58	0.37	0.45				
+ PoS left and right	0.82	0.58	0.38	0.46				
+ MSI left and right	0.84	0.62	0.37	0.46				
all	0.84	0.62	0.37	0.46				

Table 5.2 *Results for the connector features*

fact that the same connector has been used in all sentences, namely the verb 'to be'. All sentences have therefore been classified as definitions. In the *punctuation* definitions, in which two different connectors have been distinguished, using the connector phrase as a feature improves the precision scores of the pattern-based approach.

The biggest improvements of the precision are clearly obtained for the *verb* and *pronoun* definitions, for which the classifiers outperform the precision and AUC scores of the classifiers that have been trained on PoS or MSI bigrams. However, for the *verb* definitions the F-score is still below the F-score of the grammar. The complete data set contains a large variety of connector phrases as well, and as a consequence, the precision exceeds the grammar and bigrams results.

Connector phrase + context In the second and third setting related to the connector, contextual information has been included in addition to the connector phrase. The second setting contains the connector phrase, the PoS tag of the word to its left, and the PoS tag of the word to its right. The context information improves the precision for the *is* and *punctuation* data sets compared to the setting that only includes the connector phrase. For the other data sets, the precision scores with and without the context information are similar. The overall accuracy as indicated by the AUC improves for all data sets except the *verb* data set. In the third setting, the morpho-syntactic information of the left and right context has been used instead of the PoS tags. This results in better precision scores for each of the four definition types. Only on the combined data set the precision is similar with PoS tags and MSI information.

All connector information In the second and third setting, the use of PoS tags and morpho-syntactic information has been examined separately while in the last setting they are both included. In this case, the results are similar to the results obtained with the combination of connector phrase and MSI information. Apparently, the detailed morpho-syntactic information makes the more general categories superfluous for the classification. Only for the *verb* definitions there is a slight improvement when both are used.

5.2.1.3 Linguistic properties of definiendum and definiens

The linguistic properties of definiendum and definiens have been considered individually and together. Section 4.4.3.3 showed that the linguistic characteristics of defnitions and non-definitions are especially different within the *is* data set. I therefore expect that this information will be most relevant for the classification of *is* definitions. For the definiendum, the articles, adjectives and nouns have been examined. In the definiens, the usefulness of the characteristics of the first noun phrase and the type of relative clause (if present) have been investigated.

Table 5.3 summarizes the results for the linguistic settings. The characteristics of the definiendum and the definiens have been examined separately. In addition, they have been combined in one linguistic setting. The results for the sub settings within the definiendum (article, adjective, and noun) and definiens (article, adjective, and relative clause) settings can be found in Appendix G (Table G.1 and G.2).

Linguistic properties of the definiendum Individually, the noun is in all data sets the most relevant characteristics of the definiendum for the classification of definitions. However, even for this feature the AUC scores are below 0.70 for each of the definition types, which is lower than the bigram baselines. When the article, adjective and noun are combined, the scores improve, but for the individual definition types they are still low. The performance is considerably better on the aggregated data set that combines the sentences of the four definition types.

Linguistic properties of the definiens The AUC scores on the sub settings of the linguistic setting are low, especially for the *verb* and *punctuation* definitions (cf. Appendix G). The classifiers based on only this characteristic are therefore useless. In the *is* data set, the article and the type of relative clause are the best individual features. The combination of all linguistic characteristics related to the definiendum performs clearly better, both with respect to the precision and the AUC score (cf. Table 5.3). For the *is* and *pronoun* data sets, the linguistic setting even outperforms the bigram settings. The linguistic characteristics of the

	AUC	R	P	F	AUC	R	P	F
		is				**verb**		
Baselines								
grammar		0.86	0.35	0.50		0.78	0.44	0.56
PoS bigrams	0.70	0.57	0.49	0.52	0.64	0.48	0.52	0.50
MSI bigrams	0.80	0.59	0.57	0.58	0.66	0.46	0.57	0.51
Linguistic								
definiendum	0.73	0.67	0.51	0.58	0.61	0.45	0.58	0.51
definiens	0.80	0.62	0.62	0.62	0.55	0.45	0.48	0.47
both	0.84	0.67	0.62	0.64	0.63	0.44	0.54	0.49
		punctuation				**pronoun**		
Baselines								
grammar		0.88	0.07	0.12		0.75	0.09	0.16
PoS bigrams	0.77	0.63	0.15	0.24	0.69	0.42	0.16	0.24
MSI bigrams	0.79	0.56	0.18	0.28	0.66	0.34	0.14	0.20
Linguistic								
definiendum	0.63	0.46	0.12	0.19	0.63	0.48	0.17	0.25
definiens	0.64	0.44	0.16	0.23	0.71	0.48	0.17	0.25
both	0.72	0.58	0.12	0.20	0.68	0.45	0.14	0.21
		all						
Baselines								
grammar		0.82	0.16	0.26				
PoS bigrams	0.81	0.60	0.35	0.44				
MSI bigrams	0.82	0.56	0.38	0.45				
Linguistic								
definiendum	0.79	0.58	0.36	0.44				
definiens	0.80	0.58	0.39	0.47				
both	0.82	0.60	0.37	0.46				

Table 5.3 *Results for the linguistic features related to definiens and definiendum*

definiens are thus very important for these two types of definitions.

Combining linguistic properties of definiendum and definiens The last row of Table 5.3 presents the results of the classification on the basis of the linguistic information from both the definiendum and the definiens. These classifiers achieve on the one hand higher AUC scores on the *is*, *verb*, *punctuation*, and *all* data sets than the classifiers that are based on the definiendum or definiens setting alone. On the other hand, the precision scores decrease for all definition types except the *is* definitions. However, since the AUC scores give an measure for the reliability of the classifier, the classifiers with an higher AUC score are considered to be better.

5.2.1.4 Position properties

The position properties, which were presented in Section 4.4.3.4, have been split into two groups, of which the first is related to the position of the definitions and the second deals with the position of the definiendum within the document. The position of definitions within the paragraph and sentence are considered in the first group, whereas in the other setting the definiendum is investigated. More specifically, with respect to the position in the paragraph, the length of the paragraphs, and the absolute and relative position have been considered. For the position of the definiendum, the position of the term within the document is examined by investigating how often the term has been used before and after it is mentioned in the definition. In addition, the frequency of the term has been taken into account. Table 5.4 shows the results for the settings related to the position of definitions and definienda. The performance on the sub settings can be found in Appendix G (Table G.3 and G.4).

Position of definition The classifiers based on the position of the definition in the paragraph perform poorly in terms of AUC scores, which are below 0.7 for all definition types. Only for the complete data set the situation is better, but even here it does not come close to 0.8. Comparison of the sub settings (Appendix G, Table G.3) on the absolute

	AUC	R	P	F	AUC	R	P	F
		is				**verb**		
Baselines								
grammar		0.86	0.35	0.50		0.78	0.44	0.56
PoS bigrams	0.70	0.57	0.49	0.52	0.64	0.48	0.52	0.50
MSI bigrams	0.80	0.59	0.57	0.58	0.66	0.46	0.57	0.51
Position								
definition								
- in paragraph	0.63	0.59	0.47	0.52	0.53	0.4	0.51	0.45
- in sentence	0.55	0.83	0.38	0.52	0.55	0.73	0.48	0.57
- both	0.67	0.65	0.49	0.56	0.59	0.42	0.54	0.48
definiendum	0.63	0.42	0.48	0.45	0.55	0.30	0.47	0.36
all	0.67	0.51	0.46	0.48	0.62	0.47	0.53	0.50
		punctuation				**pronoun**		
Baselines								
grammar		0.88	0.07	0.12		0.75	0.09	0.16
PoS bigrams	0.77	0.63	0.15	0.24	0.69	0.42	0.16	0.24
MSI bigrams	0.79	0.56	0.18	0.28	0.66	0.34	0.14	0.20
Position								
definition								
- in paragraph	0.59	0.63	0.09	0.15	0.58	0.38	0.15	0.22
- in sentence	0.61	0.58	0.09	0.15	0.68	0.54	0.15	0.24
- both	0.70	0.57	0.12	0.20	0.68	0.45	0.14	0.22
definiendum	0.56	0.67	0.08	0.15	0.67	0.54	0.16	0.25
all	0.73	0.56	0.14	0.23	0.70	0.47	0.18	0.27
		all						
Baselines								
grammar		0.82	0.16	0.26				
PoS bigrams	0.81	0.60	0.35	0.44				
MSI bigrams	0.82	0.56	0.38	0.45				
Position								
definition								
- in paragraph	0.75	0.52	0.36	0.42				
- in sentence	0.78	0.61	0.34	0.44				
- both	0.78	0.58	0.33	0.42				
definiendum	0.77	0.59	0.33	0.42				
all	0.80	0.59	0.34	0.43				

Table 5.4 *Results for the features related to the position of definition and definiendum*

position, the relative position and the length of the paragraph reveals that the absolute position is most relevant for the classification of *is* and *punctuation* definitions, whereas the relative position setting performs better for the *verb* and *pronoun* definitions.

The start position of the definition within the sentence is especially relevant for the *punctuation* and *pronoun* sentences, where it is more important than the position in the paragraph. This is the case in the integrated data set as well, although here only the AUC score is higher, whereas the precision is slightly worse.

The results on the *is* and *punctuation* data sets improve when all information on the position of the definition is combined. For the *verb* definitions, the precision and AUC increase, whereas the recall and F-score decrease. On the *pronoun* data set, the sentence position alone performs better than the setting in which the paragraph position has been added.

Position of definiendum Although our investigations showed that there are significant differences with respect to the position of the definiendum (Section 4.4.3.4), this finding is not reflected in the experiments. Compared to the other position settings, this is the least relevant type of position information. Only for the *pronoun* definitions the information is equally relevant as the sentence position, whereas for the other data sets position setting performs worst. The AUC scores are well below 0.7 for each of the definition types. Within the sub settings related to the definiendum position (cf. Appendix G, Table G.4), the absolute, relative and balanced position settings perform similar in terms of precision and F-scores.

All position information The combination of all position information generally gives the best results of the different position settings in terms of AUC scores. However, when looking at the F-scores, both the *is* and the *verb* get better results when the position of the sentence is used. The information on the position of the definiendum, which is on its own not very relevant, clearly contributes to the improved performance for the classification of the *punctuation* and *pronoun* definitions.

The position setting improves the grammar results for the *punctu-*

	AUC	R	P	F	AUC	R	P	F
		is				**verb**		
Baselines								
grammar		0.86	0.35	0.50		0.78	0.44	0.56
PoS bigrams	0.70	0.57	0.49	0.52	0.64	0.48	0.52	0.50
MSI bigrams	0.80	0.59	0.57	0.58	0.66	0.46	0.57	0.51
Layout								
definiendum	0.52	0.15	0.57	0.24	0.54	0.13	0.69	0.23
paragraph	0.50	0.83	0.36	0.50	0.50	0.70	0.45	0.55
both	0.52	0.17	0.51	0.25	0.52	0.13	0.57	0.21
		punctuation				**pronoun**		
Baselines								
grammar		0.88	0.07	0.12		0.75	0.09	0.16
PoS bigrams	0.77	0.63	0.15	0.24	0.69	0.42	0.16	0.24
MSI bigrams	0.79	0.56	0.18	0.28	0.66	0.34	0.14	0.20
Layout								
definiendum	0.55	0.25	0.13	0.18	0.48	0.05	0.11	0.07
paragraph	0.59	0.30	0.17	0.22	0.43	0.17	0.07	0.10
both	0.64	0.44	0.16	0.23	0.42	0.18	0.07	0.10
		all						
Baselines								
grammar		0.82	0.16	0.26				
PoS bigrams	0.81	0.60	0.35	0.44				
MSI bigrams	0.82	0.56	0.38	0.45				
Layout								
definiendum	0.75	0.53	0.37	0.44				
paragraph	0.76	0.55	0.36	0.44				
both	0.77	0.54	0.37	0.44				

Table 5.5 *Results for the features related to the layout of definition and definiendum*

ation, *pronoun* and aggregated data sets. However, this is not the case for the *is* and *verb* definitions. When comparing the results to the bigram classifiers, the situation is even worse, since for four of the five data sets, the classifiers trained on position settings cannot outperform these classifiers. Only on the *pronoun* data set the performance is considerably better than the results obtained with the bigram settings.

5.2.1.5 Layout properties

In the layout setting, the type of paragraph and the layout of the definiendum are considered. The results obtained with individual settings of the layout setting are not acceptable . Most AUC scores (Table 5.5) are below 0.6, which means that these classifiers are not able to discriminate between definitions and non-definitions. The added value of this information appears only when it is combined with other properties. It is clear that the bigram settings and the grammars outperform the layout settings for all data sets.

5.2.1.6 Keyword properties

For the keyword properties, the three types of keyword scores (RIDF, TFIDF and ADRIDF) and the combination of these three scores have been used as features. The results from Table 5.6 reveal that the keyword scores alone do not provide enough information to enable a good discrimination between definitions and non-definitions. There are some AUC scores above 0.6, but this still indicates that only a poor discrimination is possible. For the complete data set, the AUC scores are better, but the precision scores are low here as well. The bigram settings outperform the keyword settings for all data sets.

5.2.2 COMPARING THE INDIVIDUAL SETTINGS

I showed in the previous section that the different features of definitions are not all equally relevant. This section investigates for each of the definition types which information is most important for the classification of definitions. To this end, five settings are compared in which all the characteristics of a certain type of definition are combined. For

	AUC	R	P	F	AUC	R	P	F
		is				**verb**		
Baselines								
grammar		0.86	0.35	0.50		0.78	0.44	0.56
PoS bigrams	0.70	0.57	0.49	0.52	0.64	0.48	0.52	0.50
MSI bigrams	0.80	0.59	0.57	0.58	0.66	0.46	0.57	0.51
Keyword								
ridf	0.57	0.41	0.42	0.42	0.53	0.31	0.50	0.38
tfidf	0.67	0.52	0.48	0.50	0.53	0.32	0.52	0.40
adridf	0.62	0.46	0.44	0.45	0.48	0.29	0.46	0.36
all	0.64	0.46	0.46	0.46	0.55	0.33	0.51	0.40
		punctuation				**pronoun**		
Baselines								
grammar		0.88	0.07	0.12		0.75	0.09	0.16
PoS bigrams	0.77	0.63	0.15	0.24	0.69	0.42	0.16	0.24
MSI bigrams	0.79	0.56	0.18	0.28	0.66	0.34	0.14	0.20
Keyword								
ridf	0.60	0.48	0.09	0.15	0.56	0.55	0.12	0.19
tfidf	0.57	0.46	0.08	0.13	0.60	0.59	0.12	0.20
adridf	0.61	0.50	0.09	0.15	0.56	0.56	0.12	0.20
all	0.62	0.46	0.09	0.16	0.59	0.56	0.12	0.20
		all						
Baselines								
grammar		0.82	0.16	0.26				
PoS bigrams	0.81	0.60	0.35	0.44				
MSI bigrams	0.82	0.56	0.38	0.45				
Linguistic								
ridf	0.76	0.55	0.32	0.40				
tfidf	0.76	0.54	0.32	0.40				
adridf	0.75	0.53	0.31	0.39				
all	0.78	0.56	0.35	0.43				

Table 5.6 *Results for the features based on the keyword scores of the definiendum*

example, for the connector setting this means that the setting includes the connector phrase, and the PoS tags and morpho-syntactic information of the words to its left and right. The five settings are the connector, linguistic, position, layout and keyword setting. The bigram settings and the results of the grammars are used as baselines against which the five settings are compared.

	AUC	R	P	F
baselines				
grammar		0.86	0.35	0.50
PoS bigrams	0.70	0.57	0.49	0.52
MSI bigrams	0.80	0.59	0.57	0.58
settings				
connector	0.71	0.66	0.50	0.57
linguistic	**0.84**	**0.67**	**0.62**	**0.64**
position	0.67	0.51	0.46	0.48
layout	0.52	0.17	0.51	0.25
keywords	0.64	0.46	0.46	0.46

Table 5.7 *Results for the* is *definitions with the individual settings*

5.2.2.1 *Is* definitions

By far the best individual setting of Table 5.7 is the linguistic setting, which improves on the precision obtained with the grammar and also outperforms the PoS and PoS+MSI bigram settings. Closer inspection of the linguistic sub settings (cf. Appendix G, Table G.1 and G.2) reveals that especially the articles used in definiendum and definiens, and the type of relative clause contribute to the good performance of this setting. In general, the information on the definiens, including the information on the presence of a relative clause, provides especially relevant information for the classification of *is* definitions. The second best setting is the connector setting. The left and right context represented by the morpho-syntactic information of the word directly to the left and to the right of the connector are quite relevant here. This setting outperforms the PoS bigrams setting, but performs worse than the MSI bigrams setting. The three remaining settings improve on the precision of the grammar, but perform considerably below both bigram

baselines. Especially the layout information seems to be a poor indicator for the classification of sentences and the AUC score obtained with this setting shows that the classifier trained on this information cannot successfully differentiate between definitions and non-definitions.

The aim of the machine learning component is to improve on the precision scores obtained with the grammar. With the linguistic setting, an improvement of 0.27 (77.1%) is obtained for this score. The price for this improvement is a decrease of the recall score with 22.1% but there is an overall gain, since the F-score improves with 28%. The connector setting improves on the F-score obtained with the grammar (14%) as well, whereas the other three settings reduce the F-score from the grammar.

	AUC	R	P	F
baselines				
grammar		0.78	0.44	0.56
PoS bigrams	0.64	0.48	0.52	0.5
MSI bigrams	0.66	0.46	0.57	0.51
settings				
connector	**0.72**	**0.54**	**0.61**	**0.57**
linguistic	0.63	0.44	0.54	0.49
position	0.62	0.47	0.53	0.5
layout	0.52	0.13	0.57	0.21
keywords	0.55	0.33	0.51	0.4

Table 5.8 *Results for the* verb *definitions with the individual settings*

5.2.2.2 *Verb* definitions

Table 5.8 shows that the connector setting is the best performing individual setting for the *verb* definitions. More specifically, it is the connector phrase that contributes highly to a good classification. The connector setting clearly outperforms the PoS and MSI bigram baselines, both with respect to AUC and precision. Whereas the linguistic setting performed very well on the *is* definitions, this is clearly not the case for the *verb* definitions. The classifier trained on the basis of this information performs similarly to the classifier trained on the PoS bigrams setting, but below the PoS+MSI baseline. For the position setting, only the

combination of all position information leads to an AUC higher than 0.6. This still indicates a poor classification, but it is similar to the PoS bigram baseline and only slightly below the MSI bigram baseline. The AUC scores are quite low for the keyword and layout settings as well, where the score is even lower than 0.6. Such low scores indicate that with the overall classifier there is no discrimination at all. This means that training a classifier on these settings does not make sense for the *verb* definitions, since it does not improve the result obtained with the grammar.

The improvement of the precision compared to the grammar is 0.17 when the connector setting is used, which means an improvement of 38.6%. However, the price for this improvement is a decrease of the recall score with 29.9% and the F-score does not change. For all other individual settings, the F-score decreases compared to the score obtained with the grammar. This means that the use of machine learning techniques does not improve the overall classification of definitions when these settings are used; the only difference is that the precision improves at the expense of the recall.

	AUC	R	P	F
baselines				
grammar		0.88	0.07	0.12
PoS bigrams	0.77	0.63	0.15	0.24
MSI bigrams	0.79	0.56	0.18	0.28
settings				
connector	**0.74**	**0.59**	**0.13**	**0.22**
linguistic	0.72	0.58	0.12	0.2
position	**0.73**	**0.56**	**0.14**	**0.23**
layout	0.64	0.44	0.16	0.23
keywords	0.62	0.46	0.09	0.16

Table 5.9 *Results for the* punctuation *definitions with the individual settings*

5.2.2.3 *Punctuation* definitions

Only three of the classifiers produce acceptable results when looking at the AUC scores (cf. Table 5.9). These are the position setting, the con-

nector setting and the linguistic setting. The other two classifiers (keyword and layout) have a AUC score below 0.7. With respect to the precision, the position setting is the best individual setting for the *punctuation* definitions. Although it clearly improves on the F-score obtained with the grammar, it nevertheless performs below the PoS and MSI bigram baselines. The difference with the PoS bigram baseline mainly lies in the recall score whereas for the MSI bigram baseline the precision is higher and the recall is similar. Within the position setting, especially the position of the definition within the sentence seems to be important. The improvement of the precision and F-score compared to the grammar results is respectively 100% and 91.7%. The second best setting is the connector setting. The left and right context are especially relevant here. This setting performs below the PoS and MSI bigram baselines as well, especially with respect to precision and F-score. The linguistic setting performs similar to the position and connector settings. The layout and keyword classifiers produce the worst results.

	AUC	R	P	F
baselines				
grammar		0.75	0.09	0.16
PoS bigrams	0.69	0.42	0.16	0.24
MSI bigrams	0.66	0.34	0.14	0.20
settings				
connector	**0.83**	**0.60**	**0.24**	**0.35**
linguistic	0.68	0.45	0.14	0.21
position	0.70	0.47	0.18	0.27
layout	0.42	0.18	0.07	0.10
keywords	0.59	0.56	0.12	0.20

Table 5.10 *Results for the* pronoun *definitions with the individual settings*

5.2.2.4 *Pronoun* definitions

Table 5.10, shows that the connector setting is by far the best setting for the *pronoun* definitions. Whereas the connector phrase itself already provides quite useful information, the performance improves even further when the left and right linguistic context information are

added. The precision improves with 166.7% compared to the precision obtained with the grammar and the F-score improves with 118.8%. The connector setting performs much better than the bigram baselines. The second best results are achieved with the position settings. Especially the start position of the definition within the sentence provides valuable information here. The classifier trained on the position setting as a whole outperforms the bigram baselines. The linguistic setting results are below the PoS bigram baseline but similar to the baseline in which morpho-syntactic information has been included. For the remaining two settings, the AUC scores are below 0.6. For the keyword setting, the F-score of the definition class has improved compared to the grammar, but the results are below both bigram baselines. The performance is worst for the layout setting, which achieves an F-score of only 0.1 and thus decreases with 37.5% compared to the grammar result.

	AUC	R	P	F
baselines				
grammar		0.82	0.16	0.26
PoS bigrams	0.81	0.6	0.35	0.44
MSI bigrams	0.82	0.56	0.38	0.45
settings				
connector	**0.84**	**0.62**	**0.37**	**0.46**
linguistic	0.82	0.6	0.37	0.46
position	0.8	0.59	0.34	0.43
layout	0.78	0.56	0.35	0.43
keywords	0.77	0.54	0.37	0.44

Table 5.11 *Results for all definitions with the individual settings*

5.2.2.5 All definitions

The obvious advantage of combining the different definition types into one data set is that there is a larger amount of data on which the classifiers can be trained, which generally results in a better classifier. The results obtained on this combined data set are shown in Table 5.11. Regardless of the setting that is used, the classifiers produce acceptable or even excellent results according to the AUC scores. For none of the four definition types this is the case when they are used separately.

The best individual setting is the connector setting. The combination of the connector phrase and its surrounding context – represented by PoS tags or morpho-syntactic information of the words – provides the best results. The classifier based on this information outperforms both the PoS bigram and MSI bigram baselines. The improvement over the precision of the grammar is 131.3% and the F-score improves with 76.9%, which means that the loss in recall is considerably smaller than the gain in precision.

The second best individual setting is the linguistic setting, which performs almost as good as the connector setting. Compared to the bigram baselines, the results are slightly better than both of them. The position information is quite relevant for the classification as well. Although the scores for this setting are slightly lower than the results obtained with the connector or linguistic settings, they are still similar to the PoS and MSI bigram baselines.

The remaining two settings perform below the PoS and MSI bigram baselines when looking at their recall scores. However, whereas the keyword scores for the four definition types individually do not provide good results, the situation changes considerably when they are used for the aggregated data set. Although this setting is still worse than the connector, linguistic and position settings, the difference is much smaller now. The layout setting, which performed worse for each of the four definition types as well, ends at the last position. The situation with this setting is very similar to the keyword setting. Just as for the layout setting, the performance of the classifier is better when trained on the combined data set instead of the four type specific data sets.

5.2.3 RANKING THE INDIVIDUAL SETTINGS

The overview for each of the definition types shows that there is a clear distinction between the relevance of the five settings. To investigate which of the settings can be considered the best setting, they have been ranked for each definition type. For the ranking, the scores have been converted to ordinal numbers (1-5) and assigned a score to each setting from best (1) to worst (5). This means that minor and major differences

between two settings are treated in the same way.

	is	verb	punct	pron	all	rank
connector	2	1	1	1	1	1
linguistic	1	2	3	3	2	2
position	3	3	2	2	3	3
keywords	4	4	5	4	5	4
layout	5	5	4	5	4	5

Table 5.12 *Ranking the individual settings on the basis of the AUC scores*

Table 5.12 shows the rankings for each of the settings. The settings are ordered on the basis of the overall rank (last column). The rankings reveal that the connector setting generally is the best individual setting. The linguistic and position setting are quite close to each other and are ranked overall as the second and third settings. The keyword setting ends at the fourth place, while the layout setting is at the last position.

5.3 COMBINED SETTINGS

The results presented so far focused on the use of one type of information. In this section, the individual settings are combined in different ways to find out whether the combination of the settings can enhance the results. Since there are five individual settings, there are 26 combined settings possible. More specifically, there are ten combinations of two settings, ten combinations of three settings, five combinations of four settings, and one combination of all settings. The complete tables with the results for all these 26 settings are presented in Appendix H. This section presents for each definition type the best combinations of two, three, four, and five settings.

5.3.1 *Is* DEFINITIONS

Table 5.13 shows the best combinations of settings for the *is* definitions. When combining two settings, the combination of the position setting and the linguistic setting gives the best results. This is the only combination that improves on the results obtained with the linguistic setting

	AUC	R	P	F
baselines				
grammar		0.86	0.35	0.50
PoS bigrams	0.70	0.57	0.49	0.52
MSI bigrams	0.80	0.59	0.57	0.58
settings				
linguistic	0.84	0.67	0.62	0.64
ling+pos	0.86	0.70	0.64	0.67
conn+ling+pos	0.86	0.71	0.67	0.69
conn+ling+pos+lay	0.87	0.70	0.66	0.68
all	0.87	0.72	0.65	0.68

Table 5.13 *Best combinations of settings for the* is *definitions*

alone. Comparison of the combinations of two settings to the baselines (Appendix H) reveals that that seven out of the ten settings outperform the PoS bigram baseline while four perform better than the MSI bigrams baseline as well. Each of these four settings includes the linguistic setting, which also proved to be the best individual setting. Compared to the grammar alone, all combinations of two settings either perform similar or better with respect to the F-scores.

The best combination of three settings, is the combination of the linguistic, position and connector settings. The addition of the connector setting especially improves the precision score compared to the combination of the linguistic and position settings. When three settings are combined, nine out of the ten settings achieve results above the PoS bigram baseline. Only the combined setting of position, layout and keyword information performs slightly below this baseline. The six settings that include the linguistic setting all outperform the MSI bigrams baseline whereas the remaining four settings perform below this baseline.

Adding a fourth setting to the combination of the linguistic, position and connector settings does not improve the results. The only combination of four settings that performs below the MSI bigrams baseline, is the setting in which the linguistic information is not included. The combination of all settings also does not improve on the results obtained with three settings.

The best combined setting for the *is* definitions is thus the combina-

tion of position, linguistic, and layout information. This combination improves on the precision and F-score of the grammar with 91.4% and 38% respectively.

	AUC	R	P	F
baselines				
grammar		0.78	0.44	0.56
PoS bigrams	0.64	0.48	0.52	0.5
MSI bigrams	0.66	0.46	0.57	0.51
setings				
connector	0.72	0.54	0.61	0.57
conn+ling	0.74	0.54	0.63	0.58
conn+ling+lay	0.74	0.54	0.64	0.59
conn+ling+lay+kw	0.75	0.55	0.64	0.59
all	0.73	0.53	0.62	0.57

Table 5.14 *Best combinations of settings for the* verb *definitions*

5.3.2 *Verb* DEFINITIONS

Table 5.14 shows that the classifiers trained on combined settings do not improve considerably over the scores obtained with the connector setting alone. The best combination of two settings is the combination of connector and linguistic information, which performs only slightly better than the connector setting. In the other three combinations that include the connector setting (cf. Appendix H), the F-score is similar to the F-score of the grammar (0.56). For the settings without connector information, the gain in precision still does not compensate the lack in recall and as a consequence, the F-scores are below the grammar F-score.

Including more than two settings barely improves the results obtained with two settings. On the contrary, for many of the combinations the results get slightly worse. The classifier trained on all settings performs has a lower AUC than the classifier trained on only the connector setting.

The best combined setting for the classifications of *verb* definitions is the combination of connector, linguistic, and layout information. This combination improves on the precision and F-score of the grammar with 45.5% and 5.4% respectively.

	AUC	R	P	F
baselines				
grammar		0.88	0.07	0.12
PoS bigrams	0.77	0.63	0.15	0.24
MSI bigrams	0.79	0.56	0.18	0.28
settings				
connector	0.74	0.59	0.13	0.22
ling+pos	0.81	0.61	0.17	0.26
ling+pos+lay	0.82	0.58	0.18	0.27
ling+pos+lay+kw	0.82	0.59	0.18	0.28
all	0.78	0.61	0.17	0.27

Table 5.15 *Best combinations of settings for the* punctuation *definitions*

5.3.3 *Punctuation* DEFINITIONS

Although the best individual results are obtained with the connector setting, the best combination of two settings does not include this setting. Table 5.15 reveals that the combination of the linguistic and position settings gives the best results. With this combination, the performance is better than both bigram baselines with respect to the AUC score while the precision is slightly below the precision of the MSI bigrams baseline. A general observation of the settings in which two information types are combined is that the AUC scores are considerably higher than for the separate settings (cf. Appendix H). In this respect, especially the combination of the layout and keyword settings – which individually both have an AUC score far below 0.7 – proves to be successful.

The addition of the layout setting to the combination of linguistic and position settings, again improves the results. However, this is only a minor improvement. The results obtained with these three settings do not improve when a fourth and fifth setting are added. When all settings are included, the results get even slightly worse.

The best combined setting for the punctuation types is thus the combination of position, linguistic, and layout information. This combination improves on the precision and F-score of the grammar with 142.9% and 116.7% respectively.

	AUC	R	P	F
baselines				
grammar		0.75	0.09	0.16
PoS bigrams	0.69	0.42	0.16	0.24
MSI bigrams	0.66	0.34	0.14	0.20
settings				
connector	0.83	0.60	0.24	0.35
conn+kw	0.83	0.60	0.25	0.35
conn+ling+lay	0.84	0.59	0.29	0.39
conn+ling+lay+kw	0.84	0.59	0.29	0.39
conn+ling+pos+lay+kw	0.84	0.60	0.29	0.39

Table 5.16 *Best combinations of settings for the* pronoun *definitions*

5.3.4 *Pronoun* DEFINITIONS

For the *pronoun* definitions, Table 5.16 shows that adding a second setting to the connector setting does not improve the results in terms of the AUC score. However, there is an improvement of the precision score to 0.27 when the linguistic setting is combined with the connector setting (cf. Appendix H). This has not been mentioned as best setting in Table 5.16, since the best combined settings here have been selected on the basis of the AUC scores.

When the linguistic, connector, and layout settings are combined, the precision score improves even further to 0.29. The other scores do not change. Combining more than three settings does not increase the performance anymore. The results with four and five settings are almost the same as the results obtained with three settings.

The best performance is thus obtained with a combination of the connector, linguistic and layout settings. The classifier trained on this information enhances the results of the grammar considerably: the precision improves with 222.2% and the F-score with 143.8%.

5.3.5 ALL DEFINITIONS

The last data set is the aggregated data set containing all types of definitions. Table 5.17 reveals that the combination of the linguistic and position setting is the best combination of two classifiers, and especially improves the precision. The addition of the connector setting as a third

	AUC	R	P	F
baselines				
grammar		0.82	0.16	0.26
PoS bigrams	0.81	0.6	0.35	0.44
MSI bigrams	0.82	0.56	0.38	0.45
settings				
connector	0.84	0.62	0.37	0.46
ling+pos	0.86	0.61	0.41	0.49
conn+ling+pos	0.86	0.65	0.42	0.51
conn+ling+pos+lay	0.87	0.64	0.43	0.51
all	0.87	0.64	0.44	0.52

Table 5.17 *Best combinations of settings for the combined data set containing the four definition data sets*

setting results in an improved recall. Combining four or five settings leads again to a slightly improved performance. When all settings are used, the precision improves with 175% compared to the precision obtained with the grammar. The F-score improves considerably as well and increases with 100%.

5.4 ADDING THE BIGRAM SETTINGS

A general observation of the results presented so far is that the AUC scores are often disappointing, especially for the individual settings. In this case, the AUC scores are often higher in the bigram baseline settings. An aspect that has not been considered yet is the combination of bigrams and other text characteristics. The classifiers presented in Section 5.2 and 5.3 are either based on only bigrams or on (combinations of) other text characteristics without bigrams.

Since the bigram baselines generally have a higher AUC score than the individual settings, experiments have been conducted to discover whether the addition of bigrams to the individual and combined settings improves the AUC and precision scores. The results with the bigrams are examined from two perspectives in this section. On the one hand, it has been investigated whether the addition of bigrams improves the results obtained with the five settings. The opposite has been done as well: the improvements when adding these settings to

the bigrams have been examined.

5.4.1 ADDING BIGRAMS TO THE INDIVIDUAL SETTINGS

The AUC scores are especially low for the individual settings (Section 5.2). The settings presented in this section correspond to the ones discussed in Section 5.2.2. More specifically, the connector, linguistic, position, keyword and layout settings have been combined with PoS and MSI bigrams to investigate whether the classification results improve. The tables presented in this section only mention the AUC and precision scores, since the focus is on the improvement of AUC and precision. The complete table containing the recall and F-scores in addition to the AUC and precision, can be found in Appendix H.

5.4.1.1 *Is* definitions

Table 5.18 reveals that the combination of the position or keyword setting with the bigrams results in an improvement of the results from both perspectives: adding bigrams improves the classfiers based on the position and keyword settings while adding the settings improves the bigram classifiers. This indicates that the bigrams and the settings complement each other and there is a mutual benefit. Adding PoS tag bigrams to the connector or linguistic setting barely improves the results that have been obtained without the bigrams, whereas the results improve from the bigram perspective. For the layout setting, the combination with the PoS bigrams improves the classifier, but the bigram

	no bigrams		PoS bigrams		PoS+MSI bigrams	
	AUC	P	AUC	P	AUC	P
bigrams			0.70	0.49	0.80	0.57
connector	0.71	0.50	0.74	0.51	0.82	0.66
linguistic	0.84	0.62	0.84	0.62	0.86	0.65
position	0.67	0.46	0.81	0.56	0.83	0.63
layout	0.52	0.51	0.68	0.47	0.80	0.58
keywords	0.64	0.46	0.77	0.55	0.83	0.62

Table 5.18 *Results for the individual settings (left) combined with PoS bigrams (middle) and PoS+MSI bigrams (right) on the* is *definitions*

setting alone gives better results than the combination of the two.

The situation is different for the MSI bigrams, where each of the settings that include this information outperform the settings without the bigrams. When the five settings are added to the MSI bigrams there is an added value as well for four of the settings. The layout setting forms an exception, since here the results do not change.

5.4.1.2 *Verb* definitions

The results for the *verb* definitions are provided in Table 5.19. The achievements of the classifiers improve when PoS tag bigrams are added to the *position*, *layout*, or *keyword* settings whereas they remain the same for the *connector* and *linguistic* settings. From the other perspective, the addition of the *connector*, *position*, and *layout* settings to the PoS tag bigrams enhances the classifiers that are based on the bigrams alone. The biggest mutual benefit is observed when the *position* setting is used.

Again, the results are different when MSI bigrams are employed. All classifiers improve when MSI bigrams are added. The *keyword* setting is the only one that does not improve the results obtained with the MSI bigrams alone. Apparently, this information is not relevant for the classification of *verb* definitions.

	no bigrams		PoS bigrams		PoS+MSI bigrams	
	AUC	P	AUC	P	AUC	P
bigrams			0.64	0.52	0.66	0.57
connector	0.72	0.61	0.72	0.59	0.75	0.62
linguistic	0.63	0.54	0.65	0.54	0.70	0.63
position	0.62	0.53	0.71	0.59	0.74	0.64
layout	0.52	0.57	0.68	0.58	0.72	0.60
keyword	0.55	0.51	0.65	0.52	0.64	0.56

Table 5.19 *Results for the individual settings (left) combined with PoS bigrams (middle) and PoS+MSI bigrams (right) on the* verb *definitions*

5.4.1.3 *Punctuation* definitions

Table 5.20 shows that the classifiers in which PoS or MSI bigrams have been used in addition to the five settings in general improve the results

	no bigrams		PoS bigrams		PoS+MSI bigrams	
	AUC	P	AUC	P	AUC	P
bigrams			0.77	0.15	0.79	0.18
connector	0.74	0.13	0.83	0.21	0.81	0.21
linguistic	0.72	0.12	0.81	0.18	0.81	0.21
position	0.73	0.14	0.83	0.19	0.82	0.21
layout	0.64	0.16	0.83	0.19	0.87	0.27
keyword	0.62	0.09	0.79	0.17	0.80	0.19

Table 5.20 *Results for the individual settings (left) combined with PoS bigrams (middle) and PoS+MSI bigrams (right) on the* punctuation *definitions*

for the *punctuation* definitions considerably, both with respect to the AUC and the precision scores. This is not surprising, given the fact that the classification on the basis of bigrams alone clearly outperforms all individual settings.

When the results are examined from the other perspective, the inclusion of keyword information in addition to the bigrams is least relevant for the classification. For the other four settings, there is a complementary gain, since the addition of these settings to the bigrams improves the results obtained with the bigrams alone, while the bigram information contributes to a better performance of the classifiers based on the individual settings.

The classifiers based on PoS and MSI bigrams perform similar, although the precision scores are generally slightly higher when the MSI bigrams are used. The classifiers that are based on keyword information perform worst, both with and without bigrams.

5.4.1.4 *Pronoun* definitions

The addition of PoS or MSI bigrams improves the AUC scores of all individual settings, as shown in Table 5.21. The best individual setting was the *connector* setting and when the PoS or MSI bigrams are added these results get even better. Also for the other settings, the addition of bigrams results in an enhanced performance of the classifiers.

The classifier based on only PoS tag bigrams does not improve when the layout information is added, while the four other settings do improve the performance. The layout setting performed worst individu-

	no bigrams		PoS bigrams		PoS+MSI bigrams	
	AUC	P	AUC	P	AUC	P
bigrams			0.69	0.16	0.66	0.14
connector	0.83	0.24	0.85	0.26	0.81	0.27
linguistic	0.68	0.14	0.74	0.19	0.74	0.20
position	0.70	0.18	0.75	0.20	0.75	0.20
layout	0.42	0.07	0.67	0.15	0.64	0.16
keyword	0.59	0.12	0.71	0.18	0.67	0.15

Table 5.21 *Results for the individual settings (left) combined with PoS bigrams (middle) and PoS+MSI bigrams (right) on the* pronoun *definitions*

ally as well. A reason for this may be the fact that often the definiendum is not included in the definition pattern, because it is in another sentence. The layout setting is partly based on the layout of the definiendum.

When the morpho-syntactic information is included in addition to the PoS tags, the AUC scores often decrease. This is also the case when only the bigrams are used. Apparently, the more general categories of the PoS tag bigrams are better suited for the classification of this type of definitions.

5.4.1.5 All definitions

The results on the aggregated data set are shown in Table 5.22. Generally, the addition of PoS and MSI bigrams to the individual settings

	no bigrams		PoS bigrams		PoS+MSI bigrams	
	AUC	P	AUC	P	AUC	P
bigrams			0.81	0.35	0.82	0.38
connector	0.84	0.37	0.87	0.44	0.87	0.45
linguistic	0.82	0.37	0.86	0.41	0.85	0.43
position	0.80	0.34	0.86	0.41	0.86	0.44
layout	0.77	0.37	0.83	0.38	0.84	0.39
keyword	0.78	0.35	0.83	0.38	0.83	0.39

Table 5.22 *Results for the individual settings (left) combined with PoS bigrams (middle) and PoS+MSI bigrams (right) on the aggregated data set with all definitions*

improves the classifiers. The biggest improvement is observed for the position setting, for which both the AUC and precision scores increase considerably when bigrams are added. Similar improvements are observed when the individual settings are added to the bigrams. All combinations outperform the purely bigram-based classifiers. For the aggregated data set, the combination of bigrams and individual settings is thus useful from both perspectives. The settings in which bigrams are included perform excellent with respect to the AUC scores, which are all above 0.8.

RANKING THE INDIVIDUAL SETTINGS

	is	**verb**	**punct**	**pron**	**all**	**rank**
PoS bigrams						
connector	3	1	3	1	1	1
position	2	2	2	2	2	2
linguistic	1	4	4	3	2	3
layout	5	3	1	5	4	4
keywords	4	5	5	4	5	5
PoS+MSI bigrams						
connector	3	1	1	1	1	1
position	2	2	2	2	2	2
linguistic	1	4	4	3	3	3
layout	5	3	3	5	4	4
keywords	4	5	5	4	5	5

Table 5.23 *Ranking the individual settings combined with bigrams on the basis of the AUC scores*

For some of the settings there is overlap with the information contained in the five settings. This is especially the case in the linguistic setting, which is heavily based on PoS and MSI information of the definiendum and definiens. It might be the case that the combination with the bigrams improves more when there is no overlap between the bigrams and the individual settings. To investigate whether this is the case, the rankings with and without the bigrams are compared.

In Section 5.4.1.5, it has been examined which setting is most relevant for the classification of definitions. The ranking of the settings

in which the bigrams are included (Table 5.23) is similar to the ranking of the settings without the bigrams, with one important difference: the linguistic setting dropped to the third place whereas the position setting climbed to the second place. Another conclusion that can be drawn on the basis of comparison of the results with and without the bigrams is that the differences between the five settings become smaller when bigrams are included (cf. Table 5.18–5.22 and Appendix H).

5.4.2 ADDING BIGRAMS TO THE COMBINED SETTINGS

In the last group of experiments, the bigram settings are added to the 26 combinations of individual settings. These experiments correspond to the ones presented in Section 5.3. The tables show the results for the best combinations of settings without using the bigram settings, i.e. the combinations that were provided in Table 5.13–5.17. The complete tables including all combined settings can be found in Appendix H.

5.4.2.1 *Is* definitions

Table 5.24 shows that the addition of the PoS and morpho-syntactic bigrams to the best combinations of settings does not improve the results to the same extent as it does for the individual settings. This is partly due to the fact that I report the results for the best combinations of settings without bigrams in this table. The improvements are bigger for the classifiers that achieved a lower performance without the bigrams (cf. Table H.1). The results obtained with the classifiers trained on one of the bigram settings in combination with two or more individual settings are all quite similar. The best AUC score that can be obtained is 0.9, while the highest precision score is 0.71.

5.4.2.2 *Verb* definitions

For the *verb* definitions, Table 5.25 reveals that it makes no difference whether the PoS or the PoS+MSI bigram setting is added. Just as for the *is* definitions, the classifiers perform similar to each other. They perform better than the settings without the bigrams, both with respect to the AUC score and with respect to the precision. The highest AUC

	no bigrams		PoS bigrams		PoS+MSI bigrams	
	AUC	P	AUC	P	AUC	P
baselines						
bigrams			0.70	0.49	0.80	0.57
linguistic	0.84	0.62	0.84	0.62	0.86	0.65
combinations						
ling+pos	0.86	0.64	0.89	0.68	0.88	0.69
conn+ling+pos	0.86	0.67	0.87	0.68	0.87	0.67
conn+ling+pos+lay	0.87	0.66	0.88	0.68	0.89	0.71
all	0.87	0.65	0.89	0.69	0.89	0.69
best results						
settings	conn+ling+pos+lay		all		ling+pos+lay	
results	0.87	0.66	0.89	0.69	0.90	0.67

Table 5.24 Is *definitions: results for the combined settings (left) with PoS bigrams (middle) and PoS+MSI bigrams (right)*

score is 0.78 whereas the best precision score is 0.68. Given the fact that the computational load is considerably higher for the PoS+MSI settings than for the PoS settings, it is remarkable that this additional computations do not lead to an improved performance.

5.4.2.3 *Punctuation* definitions

For the *punctuation* settings, the difference between the settings with and without the bigrams are bigger than for the *is* and *verb* definitions (Table 5.26). The use of bigrams means a considerable improvement for this setting. There is a clear difference between the use of PoS bigrams or PoS+MSI bigrams. The results improve gradually from no bigrams to PoS bigrams and finally to PoS+MSI bigrams. The best result is obtained when the PoS+MSI setting is combined with the linguistic, position and layout settings. The precision score is 0.29 for this classifier, which is 314.3% higher than the result obtained with the grammar.

5.4.2.4 *Pronoun* definitions

The situation with the *pronoun* definitions is different from all other data sets (Table 5.27). The PoS+MSI bigram settings perform worse than the PoS bigram settings in all classifiers for this type of definitions. Even the results of the classifiers without the bigram setting are

	no bigrams		PoS bigrams		PoS+MSI bigrams	
	AUC	P	AUC	P	AUC	P
baselines						
bigrams			0.64	0.52	0.66	0.57
connector	0.72	0.61	0.72	0.59	0.75	0.62
combinations						
conn+ling	0.74	0.63	0.75	0.65	0.75	0.64
conn+ling+lay	0.74	0.64	0.75	0.65	0.75	0.66
conn+ling+lay+kw	0.75	0.64	0.77	0.66	0.77	0.67
all	0.73	0.62	0.77	0.66	0.77	0.66
best results						
settings	conn+ling+lay+kw		conn+ling+pos		conn+pos+lay+kw	
results	0.75	0.64	0.77	0.68	0.78	0.68

Table 5.25 Verb *definitions: results for the combined settings (left) with PoS bigrams (middle) and PoS+MSI bigrams (right)*

	no bigrams		PoS bigrams		PoS+MSI bigrams	
	AUC	P	AUC	P	AUC	P
baselines						
bigrams			0.77	0.15	0.79	0.18
connector	0.74	0.13	0.83	0.21	0.81	0.21
combinations						
ling+pos	0.81	0.17	0.85	0.21	0.84	0.26
ling+pos+lay	0.82	0.18	0.87	0.23	0.88	0.29
ling+pos+lay+kw	0.82	0.18	0.86	0.24	0.86	0.28
all	0.78	0.17	0.83	0.23	0.86	0.30
best results						
settings	ling+pos+lay+kw		ling+pos+lay		ling+pos+lay	
results	0.82	0.18	0.87	0.23	0.88	0.29

Table 5.26 Punctuation *definitions: results for the combined settings (left) with PoS bigrams (middle) and PoS+MSI bigrams (right)*

either better or similar. For the PoS bigrams, the results are slightly better compared to the classifiers without bigrams, but the differences are only marginal.

	no bigrams		PoS bigrams		PoS+MSI bigrams	
	AUC	P	AUC	P	AUC	P
baselines						
bigrams			0.69	0.16	0.66	0.14
connector	0.83	0.24	0.85	0.26	0.81	0.27
combinations						
conn+lay	0.83	0.25	0.86	0.28	0.82	0.25
conn+ling+lay	0.84	0.29	0.85	0.29	0.81	0.25
conn+ling+lay+kw	0.84	0.29	0.85	0.30	0.82	0.28
all	0.84	0.29	0.84	0.29	0.79	0.25
best results						
setting	all		conn+lay		conn+ling+lay+kw	
results	0.84	0.29	0.86	0.28	0.82	0.28

Table 5.27 Pronoun *definitions: results for the combined settings (left) with PoS bigrams (middle) and PoS+MSI bigrams (right)*

5.4.2.5 All definitions

Table 5.28 shows that on the aggregated data set the bigram settings outperform the settings in which no bigrams are used. There is a slight difference between the use of PoS or PoS+MSI bigrams in terms of precision, whereas the AUC scores are most times exactly the same. The best results are obtained when all features are used, including the MSI bigrams.

5.5 CONCLUSIONS

In this chapter, I presented the results obtained with the sequential combination of a pattern-based approach and machine learning techniques. The features that have been used in the experiments can be divided in six groups. The first type of information are the linguistic bigrams (PoS tags or PoS tags and morpho syntactic information). The bigrams have been used as baseline. A third baseline is constituted by the results of the grammar.

	no bigrams		PoS bigrams		PoS+MSI bigrams	
	AUC	P	AUC	P	AUC	P
baselines						
bigrams			0.81	0.35	0.82	0.38
connector	0.84	0.37	0.87	0.44	0.87	0.45
combinations						
ling+pos	0.86	0.41	0.88	0.46	0.88	0.46
conn+ling+pos	0.86	0.42	0.89	0.46	0.89	0.48
conn+ling+pos+lay	0.87	0.43	0.89	0.47	0.89	0.47
all	0.87	0.44	0.89	0.48	0.89	0.50
best results						
setting	all		all		all	
results	0.87	0.44	0.89	0.48	0.89	0.50

Table 5.28 *All definitions: results for the combined settings (left) with PoS bigrams (middle) and PoS+MSI bigrams (right)*

The other five features settings are related to connector information, linguistic information of definiendum and definiens, position information, layout information, and keyword information. The comparison of definitions and non-definitions with respect to these features (presented in Chapter 4) revealed that the definition type plays an important role. The largest differences within the data set are observed in the *is* data set while the differences are smallest in the *verb* data set.

The experiments with classifiers based on these features revealed that the definition type to some extent influences which type of information is most relevant. Generally speaking, especially the connector, linguistic and position information are relevant for a correct classification while the keyword and layout information are less relevant. The combinations of the different settings shows that it is generally enough to combine three settings. The fourth and fifth setting often do not improve on the results obtained with the best combination of three settings, regardless of the setting that is added.

In general, the addition of the bigram settings – either PoS bigrams or PoS+MSI bigrams – improves the performance of the classifiers. This is especially the case when only one of the individual settings is used. When more settings are combined, the added value of the bigrams depends on the data set: for the *is*, *verb*, and aggregated data set, both bigram settings improve on the results and there is not much difference

between the two settings while for the *punctuation* definitions, both settings improve the results as well, but the PoS+MSI bigram settings enhance the classifiers more than the PoS bigram setting. The bigram settings do not improve the results of the *pronoun* definitions.

On the basis of the experiments, we can conclude that the machine learning techniques considerably improve the results that have been obtained with the pattern-based approach. The best results are obtained with the combination of all characteristics, including the MSI bigrams. This shows that all characteristics contribute to the correct classification of definitions, even though they were not all relevant individually. Generally, the results improve only marginally when more than three types of characteristics are used. This may indicate that a limit has been reached which cannot be further improved on the basis of text characteristics alone.

Finally, the results obtained can be evaluated also from the perspective of our starting point – the semi-automatic creation of glossaries. Our aim was to develop a tool that assists learners in the creation of such glossaries. The pattern-based glossary candidate detector presents a list to the learner that contains on average 69.7 sentences of which only 10.9 are definitions. In other words, only 1 out of 8 definition candidates is correct. The sequential approach results in on average 16.9 possible definitions per document, of which 8.5 are definitions, that is half or the retrieved sentences are correct definitions. The situation for the learner or tutor who wants to create a glossary semi-automatically has thus improved, since the list for each document gets much shorter while the relative amount of definitions increases. As a consequence, the time to create a glossary reduces.

A conclusion is the place where you got tired of thinking.

Arthur Bloch (1948)

6 Conclusions and discussion

6.1 Introduction

This thesis addressed the problem of automatic definition extraction from texts. More specifically, I have investigated which information can be used for distinguishing Dutch definitions from non-definitions automatically. In this chapter, the experiments carried out, the results achieved, and the level of success of my approach are evaluated. On the basis of the study carried out, my main contributions to the area of definition extraction are presented from the linguistic, eLearning, and development perspectives. Finally, the thesis is concluded by suggesting some future directions for research on definition extraction.

6.2 Conclusions

Definition extraction constitutes a relevant task for different applications, such as question answering, dictionary building, and glossary creation. The focus of my research has been the development of a method for creating glossaries semi-automatically to assist learners or tutors in an eLearning context. In order to support learners and tutors in finding relevant information, a broad definition of 'definition' was adopted to ensure the coverage of a large diversity of patterns.

The definition extraction approach presented in this thesis consists of two phases. In the first phase, a pattern-based approach has been proposed to match sentences that conform to a restricted set of patterns (Section 6.2.1). After the pattern-based approach, a filtering step has been applied in which machine learning techniques are used to reduce

the number of non-definitions (Section 6.2.2). The combination of the pattern-based approach and machine learning techniques has resulted in a useful tool for glossary creation. The tool detects the majority of definitions while the amount of non-definitions retrieved is acceptable (Section 6.2.3)

6.2.1 DEFINITION EXTRACTION ON THE BASIS OF PATTERNS

Definitions generally contain (at least) three elements – a definiendum, a definiens, and a connector phrase that links the definiendum to the definiens. Manual inspection of 600 definitions revealed that in the connector phrases a restricted number of verbal phrases, pronouns, and punctuation characters are employed. On the basis of this observation, a classification methodology has been proposed that distinguishes four types of definitions. The first group – the *is* definitions – contains the definitions in which the verb 'to be' connects the definiens and the definiendum while the second group – the *verb* definitions – includes all other verbal connector phrases, such as *consist of*, *mean*, and *stands for*. In the third category, a punctuation character is used as the connector (e.g. a colon) and therefore these are called *punctuation* definitions. The last group contains definitions in which either a pronominal phrase constitutes the connector part, or a pronoun phrase in combination with a verbal connector phrase is employed. Both types are referred to as *pronoun* definitions. Apart from the connector phrase, the structures employed in the definienda and the begin of the definiens are restricted as well. For these elements, a number of syntactic patterns typical for definitions can be distinguished.

On the basis of the connector phrases and the lexico-syntactic patterns from the definiendum and definiens, a pattern-based grammar has been created for each of the four definition types. The grammars are able to detect 81.3% of the definitions, regardless of the definition type. This shows that hand-crafted grammars can be used to detect a large variety of definitions. The main problems related to the detection of definitions are errors that have been made by the linguistic annotation tools and complex patterns that are rarely used in the corpus. These problems illustrate an important shortcoming of the pattern-based ap-

proach – the lack of flexibility. Only definitions that are exactly described by the regular expressions are matched.

The patterns described in the grammars matched a number of non-definitions as well, especially for the *punctuation* and *pronoun* definitions. As a consequence, only 15.8% of the sentences retrieved with the grammars are definitions. Qualitative evaluation of the pattern-based method revealed that learners and tutors in an eLearning environment appreciate the idea of a glossary creation tool, however, some people remarked that the amount of false hits with the tool is too high. From this we concluded that for definition extraction either more information than the patterns is necessary or the patterns are more complex than the ones described by the grammar. The pattern-based approach can be applied successfully to restrict the number of sentences a document contains to a set of sentences conforming to a definition pattern. However, in order to obtain a useful application, a filtering step has to be employed to reduce the amount of non-definitions.

6.2.2 DEFINITION CLASSIFICATION USING MACHINE LEARNING TECHNIQUES

The purpose of my research was to develop a method for distinguishing definitions from non-definitions automatically. I have shown that a pattern-based method alone leaves room for improvement. More specifically, such an approach extracts, in addition to 81% of the definitions, a considerable amount of non-definitions as well (84% of the extracted sentences). I have therefore investigated whether other text properties and techniques can be used to improve the results of the pattern-based approach.

The properties that have been examined were divided in six groups:

1. **Linguistic bigrams**
   (a) bigrams of part-of-speech tags (PoS)
   (b) bigrams of part-of-speech tags and morpho-syntactic information (MSI)
2. **Connector properties**: connector phrase and the linguistic categories of the tokens directly to its left and right

3. **Linguistic properties of definiendum and definiens**

   (a) definiendum: type of article, adjective, and noun

   (b) definiens: type of article, adjective, and relative clause

4. **Position properties**

   (a) definition: length of paragraph, absolute and relative position in the paragraph, absolute position in the sentence

   (b) definiendum: frequency in document, relative and absolute position within the document

5. **Layout properties**: layout of definitions and definiendum

6. **Keyword properties**: keyword score of definiendum

Machine learning techniques have been applied to examine whether these properties are suitable for distinguishing definitions from non-definitions. The data sets that have been used for training and testing consist of the sentences extracted with the pattern-based approach. As a consequence, there are separate data sets for each of the four definition types. In addition, an aggregated data set has been created in which the four individual data sets are combined. Since some of the data sets are highly imbalanced, the Balanced Random Forest classifier has been used to conduct the experiments.

6.2.2.1 The relevance of the properties

On the basis of the experiments, we can conclude that all the properties contribute to the correct classification of definitions. However, they are clearly not equally important. Although the relevance of the settings partially depends on the definition types, it is possible to draw some general conclusions. Overall, the bigram, linguistic, connector, and position features proved to be most relevant for the classification of definitions, while the layout and keyword properties only contribute to a correct classification when they are combined with other information.

Linguistic bigrams The two types of linguistic bigrams yielded different results. For the *is* definitions, the use of the more detailed MSI information instead of the PoS tags contributed to a better classification of definitions, while for the other types the difference between the types of linguistic bigrams are much smaller. For the *pronoun* definitions, the PoS bigrams are even better suited than the MSI bigrams. Both for the PoS and the MSI bigrams, it has been noticed that the bigrams that contain information on the connector phrase are most useful for differentiating between definitions and non-definitions.

Connector properties For the *verb*, *punctuation*, and *pronoun* definitions, the connector properties are most suitable for distinguishing definitions from non-definitions. Both the connector phrase and the linguistic categories of the left and right context contribute to the good classification results. The importance of the connector phrase for the classification of definitions is again stressed by this finding: the connector phrase constitutes the basis of the classification methodology, forms the core of the pattern-based grammars, and thus performs best in the machine learning experiments as well. Especially in the *verb* and *pronoun* definitions, the information on the connector and its context is considerably more important than any of the other properties. This can be explained by the fact that the connector phrases of these types are most diverse.

Linguistic properties In both the pattern-based approach and the machine learning experiments, linguistic information has been included. The linguistic features that have been used in the machine learning experiments are different from the pattern-based information in two ways. The machine learning features consider the frequencies of the linguistic elements (e.g. indefinite articles in definiendum) in definitions and non-definitions and take the combinations of the linguistic elements into account, whereas the pattern-based approach only determines which patterns can be identified in acceptable definitions. The linguistic properties are especially relevant for the classification of *is* definitions. On the basis of this finding, we can conclude that in the *is* definitions the lexico-syntactic structure differs most from non-defini-

tions.

Position properties The position features are relevant for the classification of *punctuation* and *pronoun* definitions, while they are less relevant for the *is* and *verb* definitions. Both the position of the definition and the definiendum contribute to the correct classification.

Layout properties My examination of definitions reveals that only a minority of definitions uses a specific layout. Although this is observed more often in definitions than in non-definitions, this property alone is not enough for the classification of definitions. The layout features only contribute to the classification when they are combined with other properties.

Keyword properties Although my investigation has clearly showed that the keyword scores of definitions differ from the non-definitions, it is not possible to classify definitions on the basis of this information alone. Just like the layout properties, the keyword scores are only useful for the classification when they are combined with other features.

6.2.2.2 The optimal combination for definition classification

The relevance of the different characteristics were not only investigated individually, but they have also been combined in several ways to discover the optimal combination. On the basis of these experiments, it is possible to conclude which features are most useful for the classification of definitions.

The linguistic bigrams My first conclusion is related to the use of the linguistic bigrams. I have shown that the MSI bigrams constitute a good basis for the classification of definitions, but that the addition of other text properties improves the classification results considerably. The precision scores of the best classifiers – including MSI bigrams – on the five data sets are on average 0.116 higher than the results for the MSI bigrams alone while the AUC score is 0.108 higher. This shows that the five types of properties contain relevant information that is not covered

by the linguistic bigrams alone. On the other hand, the classification that is based on the properties without the bigrams improves when bigrams are added, which shows that the features used do not cover all relevant linguistic information. However, the improvements are in this case considerably smaller: the precision improves on average with 0.042 while the AUC improves with 0.024.

Aggregating the data sets In the experiments, I have investigated for each of the four definition types individually which features are most important for the classification. This provided relevant insights in the way the distinct types are used and revealed that there are clear differences between them. It also was a useful distinction for the pattern-based approach. However, the experiments have revealed that for the machine learning filtering it is better to combine the four data sets, since the classifier that has been trained on the aggregated data set outperforms the type-based classifiers.

The best combination The last conclusion is that the combination of all features, including the MSI bigrams, generally gives the best classification results. This shows that all properties contribute to the correct classification of definitions, even though they were not all relevant individually. Generally, the results improve only marginally when more than three types of properties are used. This may indicate that a limit has been reached which cannot be further improved on the basis of text properties alone. The classifier based on all information reduces the amount of non-definitions extracted with the pattern-based approach. Compared to the pattern-based approach, the precision improves from 0.16 to 0.50 and the F-score from 0.26 to 0.56.

6.2.3 DEFINITION EXTRACTION FOR SEMI-AUTOMATIC GLOSSARY CREATION

The starting point of my research was the practical demand for a tool to automatically create glossaries. On the basis of the experiments, it is now possible to assess the level of success of my approach, that is, the sequential combination of a pattern-base method and machine learn-

ing techniques. Compared to other research on definition extraction for glossary creation, my results are better (Kobyliński and Przepiórkowski, 2008; Borg et al., 2009) or comparable (Del Gaudio and Branco, 2009). More details on the comparison to these approaches are provided in Section 6.3.3).

To make the performance of the two methods more concrete it is useful to look at the results from the perspective of a learner who wants to create a glossary semi-automatically. On average, a document from our corpus contains 13.4 definitions. The pattern-based glossary candidate detector presents a list to the learner that contains on average 69.7 sentences of which only 10.9 are definitions. The learner has to decide for almost 60 sentences that these are non-definitions. The glossary candidate detector in which a machine learning classifier has been applied after the pattern-based extraction of definition candidates presents on average 16.9 possible definitions per document, of which 8.5 are definitions. The situation for the learner or tutor who wants to create a glossary semi-automatically has thus improved, since the list for each document gets much shorter while the relative amount of definitions increases. As a consequence, the time needed for creating a glossary reduces.

6.3 DISCUSSION

6.3.1 THE EXTRACTION APPROACH

This thesis presents a definition classification approach that combines a pattern-based extraction method with machine learning techniques. This approach shows similarities to what has been presented in other research on definition extraction. Each pattern-based method includes the use of connector phrases (cf. Joho and Sanderson (2001); Saggion (2004); Walter and Pinkal (2006)). While the variety of the connector phrases is often restricted and focuses only on clear definition indicators (e.g. 'is a', 'means'), my approach addresses a larger diversity of patterns, since in the glossary creation application it is important to retrieve as many definitions as possible. Following Muresan and Klavans (2002); Walter and Pinkal (2006); Han et al. (2007), the connector

phrases have been combined with information on the lexico-syntactic structures of definitions. More specifically, regular expressions were formulated to match the structure of the definiendum and the beginning of the definiens.

An important drawback of the pattern-based approach appeared to be the fact that the connector phrases are often used in non-definitions as well. Different solutions have been proposed to solve this problem. Joho and Sanderson (2001) and Han et al. (2007) complemented their pattern-based approach with a method in which a number of text characteristics are included in a linear formula. The optimal weights of the different elements are determined on the basis of the definitions. An alternative way of including different text properties for classification is the use of machine learning techniques (cf. Fahmi and Bouma (2006); Del Gaudio and Branco (2009)). The experiments from Fahmi and Bouma (2006) for the classification of *is* definitions show that machine learning techniques can be successfully applied to definition extraction. Therefore, it was decided to use a sequential combination consisting of a pattern-based extraction phase and a machine learning filtering step.

6.3.2 THE FEATURES

Our main contribution to the research on definition extraction involves the use of a wide variety of text properties for the extraction of definitions. Section 6.2.2 presented an overview of the properties that have been considered.

Linguistic bigrams Word *n*-grams have been used in various other experiments on definition extraction (Androutsopoulos and Galanis, 2005; Fahmi and Bouma, 2006). The most frequent word *n*-grams often correspond to the connector phrases that have been distinguished in the pattern-based approach. This information has in my experiments been included in the *connector* properties. Instead of word *n*-grams, bigrams of part of speech tags and morpho-syntactic information have been used. While we were the first to employ such linguistic *n*-grams, other researchers have followed my approach and started to use them

as well (Kobyliński and Przepiórkowski, 2008; Del Gaudio and Branco, 2009).

Connector properties The experiments from (Androutsopoulos and Galanis, 2005) revealed that the connector phrases are similar to the most frequent word *n*-grams. My use of connector information is also related to the weighted phrases (e.g. 'is a') from the pattern-based approach of Joho and Sanderson (2001). Important differences are that the connector phrase is separated in my experiments from its context and that the linguistic categories of the surrounding words have been used instead of the actual words. For example, the phrase 'is a' is in my approach represented by a feature for the connector, 'is', and a feature for the linguistic category of the context of the connector, which is an indefinite article.

Linguistic properties The linguistic properties that have been employed are partly based on the features used by Fahmi and Bouma (2006), who looked into the articles in definiendum and definiens and the nouns in the definiendum. The use of relative clauses as a feature has not been used in machine learning before, although Muresan and Klavans (2002) have employed this information in their pattern-based extraction approach. In addition to these linguistic features, the adjectives used in definiens and definiendum have been considered. The use of linguistic properties has been extended in my approach to other types of definitions, whereas Fahmi and Bouma (2006) focused only on *is* definitions.

Position properties The position of the definition within the document has been taken into account before within the pattern-based approach (Joho and Sanderson, 2001) and as a machine learning feature (Fahmi and Bouma, 2006; Blair-Goldensohn et al., 2004). In my features, attention has been paid to the position of the definition as well. While Joho and Sanderson (2001) and Fahmi and Bouma (2006) used the absolute position of the sentence within the document, I have considered the (absolute and relative) position within the paragraph because our documents are longer.

An innovative characteristic that has been included in my experiments involves the use of the position of the definiendum within the document. In this respect, the use of the definiendum frequency in the document, and the absolute and relative position of the definiendum compared to other occurrences of the term in the same document have been investigated. It has been shown that the relative position of the definiendum is on average lower within the definitions, especially within the *is* definitions. However, for the classification the information on the position of the sentence is more relevant.

Layout properties The use of layout information has not been investigated before. This information proved to be especially useful when it is combined with other information.

Keyword properties Another innovating feature involved the use of keyword scores. Although the results that have been obtained with this information alone are worse compared to the other types of information that have been used, I have shown that this information can be employed to improve the performance obtained with other settings.

6.3.3 THE RESULTS

It is difficult to compare the performance of my approach to other results, since there are several factors that influence the classification. The most important differences are related to the data sets, the variety of patterns distinguished, and the application. My research initiated within the LT4eL project, in which for eight languages a glossary candidate detector had to be developed. While for half of the languages only a pattern-based approach has been implemented, four alternative approaches – including my method – to the classification of definitions have been examined (Kobyliński and Przepiórkowski, 2008; Borg et al., 2009; Del Gaudio and Branco, 2009). Since we all focused on the same application (glossary creation), used comparable data sets (in different languages), and distinguished the same definition types, comparison to these approaches is the best option.

	AUC	R	P	F
All definition types				
Our results	0.89	0.63	0.50	0.56
Kobyliński and Przepiórkowski (2008)	0.85	0.69	0.21	0.33
Only *is* definitions				
Our results	0.90	0.72	0.67	0.70
Borg et al. (2009)		0.51	1.00	0.68
Del Gaudio and Branco (2009)	0.94			0.74

Table 6.1 *Definition extraction for glossary creation: a comparison of four different approaches*

Kobyliński and Przepiórkowski (2008) used only machine learning techniques – a Balanced Random Forest classifier – to distinguish definitions from non-definitions. My method, in which the machine learning step is preceded by a pattern-based one, clearly outperforms their approach, especially with respect to the precision scores. Borg et al. (2009) applied a combination of Genetic Programming and Genetic Algorithms for the classification of *is* definitions. The main problem of their approach is the balance between precision and recall: all the retrieved definitions are correct, but half of the definitions are not detected. This conflicts with my goal to maximize the recall. In my approach, the balance between precision and recall is considerably better. The results from Del Gaudio and Branco (2009) in which an alternative balancing algorithm (SMOTE) has been used are slightly better than my results. However, it should be noted that they did not include the results of the pattern-based step in their performance scores. When evaluating only the machine learning step of my approach, an F-score of 0.75 is obtained.

6.4 MAIN CONTRIBUTIONS

My dissertation provides relevant insights on definitions and the extraction of definitions from several perspectives. The main contributions are presented from the linguistic perspective, the eLearning perspective, and the development perspective.

6.4.1 LINGUISTIC PERSPECTIVE

From the linguistic perspective, the main contribution involves the investigation on the characteristics of different definition types – the *is*, *verb*, *punctuation*, and *pronoun* definitions. Especially the application of definition extraction to the *pronoun* definitions was an unexplored area. I have shown that linguistic properties are of crucial importance for the classification of all definition types. The two properties that are related to the linguistic structure of the definitions – the *connector* and the *linguistic* features – proved to be the most informative ones. In addition, the linguistic bigrams performed good for the different definition types as well. The differences between definitions and non-definitions with respect to the linguistic properties of definiendum and definiens are most prominent in the *is* definitions.

6.4.2 ELEARNING PERSPECTIVE

The starting point of my research was the need for a glossary creation tool to be used in an eLearning environment. To this end, this thesis has presented a method to extract definitions on the basis of documents. To assist the learner or tutor as much as possible, the definition extraction tool addresses a large variety of definition patterns. In addition to *is* definitions, the focus has been on *verb*, *punctuation*, and *pronoun* definitions as well. Quantitative evaluation of the tool has revealed that the majority of definitions in a document is detected, while the amount of non-definitions is acceptable (cf. Section 6.2.3). From the eLearning perspective, I have thus contributed by developing a practical tool to enhance the learning process that can be implemented in a Learning Management System (e.g. BlackBoard, WebCT).

6.4.3 DEVELOPMENT PERSPECTIVE

On the basis of the results obtained in my study on definition extraction, some insights can be provided to inspire future research in this area. The combination of a pattern-based approach and machine learning proved to be a good method for the extraction of definitions. In the machine learning approach, it is important to use an algorithm that is

able to deal with imbalanced data sets, like the balanced bagging algorithm that has been implemented into the Balanced Random Forest classifier. As for the features, the linguistic bigrams in which part-of-speech tags and morpho-syntactic information are combined provide a good starting point for the classification of definitions. In this respect, the part of the sentence containing the connector phrase and its context is most relevant. The addition of connector, linguistic, and position properties to the bigrams improves on these results. Including the keyword and layout scores is not worth the effort, since they contribute only marginally to the classification of definitions.

6.5 FUTURE RESEARCH

In any research there is always room for improvement and the work described in this thesis is no different. While the primary focus has been on investigating the contribution of different types of information to the classification of definitions, there remain some other directions that deserve to be explored in more detail.

Soft matching techniques A disadvantage of the pattern-based approach was its low flexibility to cope with errors and variation. It is not tolerant to noise in training data, and cannot recognize definition patterns that are not explicitly described in the grammar. Since a high recall was important within the glossary creation context, it was necessary to create very specific patterns. The manually crafted grammars are based on a 400,000 word corpus that contains 600 definitions and although the evaluation on our test corpus revealed that the grammar patterns covered most of the definitions, it might be the case that in a different corpus additional patterns are used. Especially within the group of *verb* definitions there may be room for improvement. The use of soft matching techniques (Cui et al., 2007) might help in this respect. Such techniques could be used to compute the degree of match between the test sentences and the patterns described in the grammar using a probabilistic model.

Addressing imbalanced datasets The use of different types of classifiers is another direction for future research. Relevant work in this direction that can provide fruitful insights for future work involves machine learning experiments carried out by the Portuguese partner from the LT4eL project in this area (Del Gaudio and Branco, 2009). They compared different algorithms to address the imbalanced dataset problem of which the SMOTE algorithm (Chawla et al., 2002) seems to be the most promising. When I started my work, only the balanced bagging method had been applied to definition extraction. Instead of using the balanced bagging method, the SMOTE algorithm could be implemented to investigate whether the results can be improved further.

Borg et al. (2009) employed Genetic Algorithms to address the problem of definition extraction. One of the aspects of these algorithms is that they assign a weight to each of the features. Something similar has been done in the linear formulas used by Joho and Sanderson (2001); Han et al. (2007). The balanced random classifier that has been used in my experiments does not include feature weights. Enriching the classifier with a feature weighting procedure could improve the results.

Features from external sources The features used in my approach are restricted to properties that can be extracted from the document itself. The use of information from external resources to improve the classifiers is worthy to be explored, since they may contain valuable information for the classification as well. Prager et al. (2002) investigated the use of WordNet hypernyms, but concluded that this is not possible for all definitions, since many terms are not covered in WordNet. However, it might be nevertheless useful to include the information for the terms that do exist in WordNet. Another direction that deserves attention in this respect is the use of similarity measures to compute the relationships between the definiendum and the noun phrases from the definiens. When the definiendum and definiens are closely related to each other (e.g. *HTML* and *markup language*), it is more likely that the sentence is a definition than when the two elements are not related to each other (e.g. *HTML* and *example*).

Different features per definition type In my experiments, the same features have been used for each of the definition types. However, I noticed that some features are not relevant for all the types. They sometimes even decreased the classification performance. It would be interesting to investigate whether the classification results can be improved when only the features that increase the results are considered.

Appendices

A

LT4eLAna DTD

The corpus documents conform to the LT4eLAna DTD, which is contained in this appendix. The LT4eLAna DTD is an enriched version of the XCES DTD for linguistically annotated corpora (Ide and Suderman, 2002).

```
<!--                                                              -->
<!--                Corpus Encoding Standard                      -->
<!--                                                              -->
<!--                           CES                                -->
<!--                                                              -->
<!--         Encoding conventions for annotated data              -->
<!--                                                              -->
<!--         Modified for the LT4EL project                       -->
<!--         This version covers the purely linguistic            -->
<!--         which is used in application contexts                -->
<!--                                                              -->

<!--
Original Date: 1996/08/05 19:07:30
Original Revision: 1.11

This is a modification of the original CES DTD designed for use with
the LT4EL learning objects. Created by Lothar Lemnitzer, Adam
Przepiorkowski and Kiril Simov.

Version 3.1 -->

<!--               Global attributes                    -->

<!ENTITY % a.global '
        id              ID              #REQUIRED
        xml:lang        CDATA           #IMPLIED
        lang            CDATA           #IMPLIED
        rend            CDATA           #IMPLIED'       >
```

```
<!ENTITY % a.ana '%a.global;
        type            CDATA           #IMPLIED
        wsd             CDATA           #IMPLIED'       >

<!ELEMENT LT4ELAna      (par+)                          >
<!ATTLIST LT4ELAna
        id              ID              #IMPLIED
        xml:lang        CDATA           #IMPLIED
        lang            CDATA           #IMPLIED
        rend            CDATA           #IMPLIED
        type            CDATA           #IMPLIED
        wsd             CDATA           #IMPLIED
        version         CDATA           #IMPLIED        >

<!ELEMENT par           (s | tok)+                      >
<!ATTLIST par           \%a.ana;
          name          CDATA           #IMPLIED        >

<!ELEMENT s             (chunk | tok | markedTerm | definingText)+ >
<!ATTLIST s             \%a.ana;                        >

<!ELEMENT chunk         (chunk | tok | markedTerm)+      >
<!ATTLIST chunk         \%a.ana;
          category      CDATA           #IMPLIED        >

<!ELEMENT tok           (#PCDATA)                       >
<!ATTLIST tok           \%a.ana;\%temp.attrs;
          sp            (y|n)           "n"
          name          CDATA           #IMPLIED
          class         CDATA           #IMPLIED
          base          CDATA           #IMPLIED
          ctag          CDATA           #IMPLIED
          msd           CDATA           #IMPLIED        >

<!-- Definition of the attributes:
  sp    - determines whether after the token there was a space or
          not (it makes sense only if spaces are deleted after
          the tokenization;
  name  - the address of the tok element before the conversion from
          Basic XML into LT4ELAna XML. For the element par it plays
          the same role
  class - General classification of the tokens in the text: word,
          number, punctuation etc
  base  - The base form for the wordform (token)
  ctag  - Part of speech info
  msd   - morphosyntactic description                          -->

<!-- The following elements are specific to the project. They model
the keyword and defition elements to be annotated -->

<!ELEMENT markedTerm    (chunk | tok | markedTerm)+     >
```

```
<!ATTLIST markedTerm    \%a.ana;
        kw              (y|n)           "n"
        dt              (y|n)           "n"
        status          CDATA           #IMPLIED
        comment         CDATA           #IMPLIED        >

<!ELEMENT definingText  (chunk | tok | markedTerm)+     >
<!ATTLIST definingText
        id              ID              #IMPLIED
        xml:lang        CDATA           #IMPLIED
        lang            CDATA           #IMPLIED
        rend            CDATA           #IMPLIED
        type            CDATA           #IMPLIED
        wsd             CDATA           #IMPLIED
        def_type1       CDATA           #IMPLIED
        def_type2       CDATA           #IMPLIED
        def             IDREF           #IMPLIED
        continue        CDATA           #IMPLIED
        part            CDATA           #IMPLIED
        status          CDATA           #IMPLIED
        comment         CDATA           #IMPLIED        >
```

B

Sentences not covered by the grammar

This appendix contains the sentences that are not covered by the grammars presented in Chapter 3. The groups of errors that are distinguished correspond to the categories mentioned in Section 3.7. The words or phrases that are problematic for the grammars have been marked in bold.

Is DEFINITIONS

Development corpus (16)

- tagger errors

    1. **Postscript** is een taal waarmee de opmaak van een pagina beschreven kan worden.
    2. Het programma **xpaint** is een tekenprogramma dat geschikt om grotere bitmaps en ook kleurenplaatjes, z.g. pixelmaps te maken.
    3. **Internet** is een netwerk van netwerken.
    4. Een symbolische **link** (ook wel eens zacht genoemd) is een indirecte verwijzing naar een andere file of directory.
    5. De Korn **shell** (geschreven door Korn) is een uitbreiding van de Bourne shell met o.a. ook een history mechanisme.
    6. **tr** is een filter dat tekens kan vertalen in andere tekens.

7. **grep** is een programma dat een tekststroom leest en de regels waarin een bepaald stuk tekst voorkomt eruit filtert.
8. **sort** is een programma om files te sorteren, d.w.z. op een bepaalde volgorde te zetten.
9. **eWatch** is een waarnemingsplatform dat erop is gericht meer inzicht te verwerven in de vernieuwing en verande ring van het Europese onderwijs.

- sentence start

1. **een** zelfstandig naamwoord is een 'woord, dat met 'die' of 'dat' gecombineerd kan worden, (meestal) een enkelvouds- en een meervoudsvorm heeft en een zelfstandigheid (in de ruimste zin) aanduidt.'
2. **een** proces is een "zwarte doos" waar iets in gaat en iets uitkomt (figuur 2.1).
3. **e-Learning** is het gebruik van nieuwe multimediatechnieken en internet om de kwaliteit van het leren te verbeteren door middel van het vergemakkelijken van de toegang tot hulpbronnen en diensten én door uitwisseling en samenwerking op afstand

- complex patterns

1. LaTeXis **noch** een DeskTop Publishing-pakket **noch** een tekstverwerker, **maar** een zet-systeem.
2. Een directory is **intern in het Unix systeem** een lijst van files en (sub)directories.
3. Z39.50 is een internationaal **zoek- en opsporingsprotocol** (ISO 23950, 1998) dat het doorzoeken van heterogene databases en de opsporing van data via één gebruikersinterface mogelijk maakt.
4. Levenslang leren is een " vormingsproces gedurende het hele leven, met als doel kennis, vaardigheden en deskundigheid te optimaliseren, binnen een persoonlijk, burger-, maatschappelijk en/of werkgerelateerd perspectief ".

Test corpus (11)

- tagger errors

    1. **Markup** is het gebruik van code om de browser, een programma waarmee HTML-documenten bekeken kunnen worden, te vertellen hoe de inhoud van het document weergegeven moet worden en naar welke bestemming de hyperlinks moeten leiden.
    2. SMTP **of** simple mail transfer protocol is een mail service welke is gebaseerd op het FTP-model.
    3. Een domein-naam (domain name) is het **web-adres** zoals je dat normaal gebruikt in je web-browser.
    4. UDP of user **datagram** protocol is een transport protocol welke niet is gebaseerd op een verbinding.
    5. SGML ("Standard Generalized Markup Language") en XML ("Extensible Markup Language") **zijn** standaarden waarmee de structuur van documenten kan worden vastgelegd.
    6. Data islands **zijn** blokken xml content in een HTML-pagina.
    7. RDF en CDF **zijn** toepassingen die ontwikkeld zijn met XML.
    8. XML Schema is zelf **XML**, ondersteunt allerlei gegevenstypen en staat ook toe om zelf types te definiëren, en is bovendien zeer flexibel.

- complex patterns

    1. MONOGRAFIEËN zijn **min of meer** specialistische boeken over een relatief afgebakend onderwerp.
    2. DTD is **al sinds 1983** een standaard die gebruikt wordt voor document definities voor de Standard Generalized Markup Language (SGML).

- patterns that should be added to grammar

    1. Een "complex type" **daarentegen** is een type dat een element definieert dat attributen en/of sub-elementen bevat.

Verb DEFINITIONS

Development corpus (29)

- tagger errors

  1. Afzonderlijke waarden uit een tabel die als grafiek worden weergegeven, noemt men **gegevensmarkeringen**.
  2. Een verzameling gegevenspunten die afkomstig is uit dezelfde kolom of rij heet een **gegevensreeks** (serie).
  3. De environment **itemize** wordt gebruikt voor eenvoudige lijsten (zie fig. 2.1).
  4. De environment **enumerate** wordt gebruikt voor lijsten met genummerde regels (zie fig. 2.2).
  5. Het commando **mbox{...}** zorgt ervoor, dat het woord helemaal niet afgebroken wordt.
  6. Als we de werkfile uit de versiefile willen halen, bijv. omdat we de werkfile weggegooid hebben of omdat we een oude versie nodig hebben, dan heet dat **check-out**.
  7. e-learning veronderstelt het **gebruikmaken** van nieuwe multimediatechnologieën en het internet om daarmee onderwijs te innoveren
  8. **De** sociaal-communicatieve competentie omvat niet alleen de mondelinge-en schriftelijke vaardigheid van de eTutor, het gaat ook om kennis van en inzicht in groepsprocessen.
  9. Elektronische tijdschriften kunnen gedefinieerd worden als elke **periodiek**, tijdschrift e-zine, nieuwsbrief of artikelenreeks die via Internet beschikbaar is.
  10. Met betrekking tot digitale inhoud, betekent **interoperabiliteit** dat het zo breed mogelijk herbruikbaar moet zijn, te transporteren door verschillende netwerken, systemen en organisaties en zo duurzaam mogelijk.

- complex patterns

1. Vooralsnog **wordt hier gesteld dat er sprake is van** eLearning als leerprocessen worden gerealiseerd en begeleid binnen een elektronische leeromgeving.
2. Het begrip "agent" **is op dit moment niet helemaal duidelijk gedefinieerd maar wordt meestal gebruikt voor** een programma of robot die informatie verzamelt of een andere dienst verricht zonder de directe aanwezigheid van de gebruiker.
3. Het onderscheid tussen toegangspoorten en portalen (gateways en portals) is nogal vaag, **maar over het algemeen zal** een toegangspoort bestaan uit groepen geannoteerde links naar andere websites, die doorgelicht zijn door de makers van de gateway.
4. **In het ideale geval** betekent web-enabled, dat de docent materiaal ontwikkelt dat hij probleemloos op een webserver kan zetten.

- no connector verb

1. Een vaste spatie **voorkomt** dat een regel tussen twee woorden wordt afgebroken.
2. Een voettekst **wordt afgedrukt in** de ondermarge van iedere pagina van een sectie.
3. Werkbalken **laten U toe** op een snelle manier allerlei taken uit te voeren.
4. Een celadres **geeft** de plaats **aan** van een cel in het werkblad.
5. Het commando \\ of `\newline` **breekt** de regel **af** zonder een nieuwe alinea te beginnen.
6. Het commando * **breekt** een regel **af**, maar er mag op die plaats niet op een nieuwe pagina begonnen worden.
7. De tabular-environment **is voor** het zetten van tabellen, waarbij LaTeXautomatisch de benodigde kolombreedtes berekent en waarin ook uitvullen en hulplijnen toepasbaar zijn.
8. Een modem **vertaalt** de digitale signalen van de computer in analoge signalen (geluid) die verwerkt kunnen worden door de telefoonlijnen.

9. Het Z39.50 protocol **geeft** zoekers de mogelijkheid in een catalogus of bibliografisch bestand te zoeken, zonder de verschillende zoekstructuren te kennen die door de verschillende software leveranciers worden geleverd.
10. Een thesaurus **toont** relaties zoals hiërarchie en gelijkwaardigheid tussen gebruikte termen.
11. Portalen **richten zich** er in het algemeen **op** om diensten te bieden aan hun gebruikers als aanvulling op hun verzameling links.
12. Een cognitief gereedschap **versterkt** de intellectuele vermogens van mensen en breidt deze vermogens uit (Saljö, 1996).
13. Metadata **geven** een korte beschrijving van het materiaal, zoals de auteur, publicatiedatum, ' keywords ' en, in het geval van e-learning, bijvoorbeeld didactisch niveau.

- other problems

1. De **"x"** permissie op een directory betekent dat je "door de directory heen mag gaan", d.w.z. als je een de naam van een file in de directory weet dan kun je die filenaam gebruiken.
2. Een niet-inhoudelijke samenvatting wordt meestal een abstract genoemd**;**

Test corpus (16)

- tagger errors

1. **Scripts** kun je gebruiken om extra mogelijkheden aan HTML-documenten toe te voegen.
2. Een belangrijke tool waarmee XML documenten gelezen kunnen worden, wordt een XML **parser** genoemd, een meer formele naam is XML processor.
3. **Well-formed**, betekent vrij vertaald welgevormdheid en geeft aan of een XML document syntactisch correct is.

- complex patterns

1. De discoursanalyse impliceert **bijvoorbeeld niet alleen** een bepaalde 'technische' aanpak en uitvoering van het onderzoek, **maar** komt tevens voort uit bepaalde theoretische aannames over de sociale functie van maatschappelijke discoursen, i.e. de veronderstelling dat deze onze blik op de wereld bepalen (zie Philips/Hardy 3 - 11)
2. PNG is **als patent- en rechtenvrije tegenhanger van GIF van CompuServe ontwikkeld en** bedoeld om GIF op te volgen als standaard voor afbeeldingen op het Internet.

- no connector verb

1. Hoorcolleges **bieden** je een logisch geordend overzicht van de belangrijkste kennis, inzichten en vaardigheden die je aan het eind van de cursus geacht wordt te beheersen.
2. Het JPEG-formaat **gebruikt** een compressiemethode waarbij de bestandsgrootte van een foto wordt verkleind door het selectief verminderen van de details van de afbeelding en door het overbrengen van de afbeeldinggegevens naar een formaat dat beter geschikt is om te worden gecomprimeerd.
3. Een parser **ontleedt** een XML-document en controleert of het document conform een bepaald documentmodel is opgesteld.
4. Een non-validating parser **controleert** het XML-document enkel op well-formedness.

- other problems

1. De Dublin Core metadata set versie **1.1** bestaat uit 15 elementen voor het beschrijven van elektronische informatie.
2. XSLT wordt hoofdzakelijk gebruikt **voor voor** de transformatie van data van de ene XML structuur naar een andere.
3. **XSLT-documenten** worden stylesheets genoemd.

- patterns that should be added to grammar

1. De Dublin Core standaard **wordt gehanteerd voor** de beschrijving van metadata bij webpublicaties van de overheid.

2. DNS of domain name service **zorgt voor** de koppeling tussen een domain name (domein naam) met een ip-adres.
3. TCP of transmission control protocol **zorgt voor** een logische verbinding tussen de twee eindpunten binnen het netwerk.
4. Het opvragen van informatie via het internet wordt **ook wel** downloaden genoemd.

Punctuation DEFINITIONS

Development corpus (13)

- tagger errors

  1. **vIRC** (Een IRC client geschreven in Visual Basic, gratis en geschikt voor beginners.
  2. 400 **Bad request**: je hebt een fout of onvolledig adres gegeven
  3. 503 **Service unavailable**: als de server niet weet wat hij met je verzoek moet doen dan krijg je deze boodschap, een reden kan zijn dat er te veel verkeer is naar deze server

- comma or dash as connector

  1. Al de computers van de ISP's, permanent met elkaar verbonden (de servers of hostcomputers), hebben een uniek nummer**,** het IP-adres.
  2. Deze is gebaseerd op XSLT**,** een transformatietaal die gebruikt wordt voor het reorganiseren, toevoegen en verwijderen van tags en attributen.
  3. Een IRC-gebruiker gebruikt meestal een IRC-client**,** een programma dat ontworpen is om verbinding te maken met een of meerdere IRC-netwerken.
  4. Communicatie tussen deze duizenden netwerken en miljoenen computers is mogelijk via het TCP/IP- protocol**,** een serie technische afspraken over de data-uitwisseling.
  5. In 1979 wordt Usenet opgezet**,** het platform van nieuwsgroepen.

6. De sleutel om dit te bereiken ligt bij standaards **-** gecodificeerde regels en richtlijnen voor de creatie, omschrijving en het beheer van digitale bronnen (zie Reinventing the Wheel D-Lib Magazine Jan 2002 voor meer informatie).

- reverse order

1. Deze overeenkomst is opmerkelijk, doch niet verwonderlijk, gezien het feit dat ook Engeland koerst op 50 % participatie in het ho, wat ook daar impliceert dat een grotere diversiteit aan studenten zal instromen (widening participation).
2. In de regel zal elke opleiding e-learning in combinatie met contactonderwijs bevatten: 'blended learning'.

- other

1. De marges (**kantlijnen**)
2. **Radardiagram:Deze** is bijzonder geschikt om de verbanden tussen verschillende gegevens aanschouwelijk te maken.

Test corpus (1)

- complex pattern

1. **De beste omschrijving voor resolutie is eigenlijk:** de dichtheid van de punten per gegeven oppervlak.

Pronoun DEFINITIONS

Development corpus (17)

- tagger errors

1. Dit noemen we **afspatiëren**.
2. **Word** noemt deze tekst aantekeningen.
3. '**overzichts-weergave**' waarin u de tekst van al uw dia's onder elkaar ziet.
4. De belangrijkste mode is de paragraph **mode** waarin tekst gewoon in alinea's gezet wordt.

5. Dit noemt men **bookmarken**.
6. We spreken dan van een **link**.
7. Van tijd tot tijd - meestal wanneer een stuk werk af is - wordt de inhoud van de werkfile in de versiefile gestopt; dit heet **check-in** .

- complex patterns

1. Het **is** speciaal ontworpen **voor** het zetten en drukken van mathematische teksten en formules.
2. Dan is er ook nog het commando \chapter $*$ { ...} **, dat ervoor zorgt** dat het hoofdstuk niet genummerd wordt en dat het niet in de inhoudsopgave verschijnt.
3. Men noemt deze platformen **o.a.** 'leer management systemen' ('LMS-en'), 'educatieve platformen', 'e-learning platformen' of 'teleleerplatformen'.

- no connector verb

1. Dit programma **biedt** een aantal interessante mogelijkheden om je presentatie te ondersteunen.
2. Het **stelt** de auteur **in staat** zijn publicaties op eenvoudige wijze en met gebruik van een van te voren opgegeven structuur, met boekdruk-kwaliteit te zetten en af te drukken.
3. Dit zijn, via je browser, **te raadplegen** webpagina's waarop je databanken kan consulteren die de inhoud van het Internet catalogeren.
4. Het omgekeerde is tail - < = n, dat de laatste < = n regels geeft, of tail + < = n dat vanaf regel < n begint.
5. Deze **leggen** een verbinding tussen computers en netwerken zonder dat er fysieke verbindingen nodig zijn.
6. Dit **stelt** twee of meer mensen **in staat** om vanaf verschillende locaties elkaar te zien en te horen, en soms om samen te werken aan hun Pc's.
7. Dit systeem **reguleert** de toegang tot online databanken.

Test corpus (8)

- no connector verb

    1. Deze artikelen **zijn gebaseerd op** meer of minder oorspronkelijk bronnenonderzoek en presenteren aan de hand van een vraagstelling een interessante interpretatie over een meer of minder nauw afgebakend onderwerp.
    2. Hierin **heeft** de docent een verzameling van wetenschappelijke teksten **opgenomen** om te bestuderen.
    3. Deze kernzin **kondigt** zo compact en compleet mogelijk het thema of de hoofdgedachte van de alinea aan of vat zo kernachtig mogelijk de hoofdgedachte samen.
    4. Hierbij **bouw** je je presentatie op naar het model van een ui, bestaande uit een kern en daar omheen verschillende schillen.
    5. In zo'n werkstuk **worden** de verschillende interpretaties die historici (of theoretici) in de afgelopen decennia ten aanzien van een bepaald vraagstuk hebben **geformuleerd**, in hun onderlinge verband en in de context van hun tijd systematisch beschreven, vergeleken en beoordeeld.

- complex pattern

    1. Alle teksten die je schrijft moeten oorspronkelijk zijn, dat wil zeggen**:** in je eigen woorden zijn geschreven en een neerslag bieden van je eigen bevindingen en interpretaties.
    2. Een document **dat hieraan voldoet,** is well-formed.
    3. Deze documenten zijn **niet alleen** well-formed, maar zijn eveneens volgens een bepaalde DTD of XML Schema opgesteld.

C

Weka input: arff file

We have used the Weka machine learning package for our experiments (Witten and Frank, 2005). Our LT4ELAna XML files have therefore been converted to arff files. An `arff` file consist of three parts. In the first part, the 'relation' indicates what the document is about. The relation can be anything and has no influence on the training or testing process. The second part contains a description of the features that are used in the instances. For each instance, a name is given and the type of input is indicated (e.g. string, integer). The third part contains a list of instances, each including all attributes described in the second part. The order of the features has to match the order and format indicated in the second part. This appendix contains a fragment of the `arff` file for the *is* definitions.

```
@relation is_definitions

@attribute mt_uni_ctagmsd string
@attribute def_uni_ctagmsd string
@attribute aft_conn_ctag_unimsd string
@attribute mt_uni_ctag string
@attribute def_uni_ctag string
@attribute aft_conn_ctag_uni string
@attribute conn string
@attribute conn_befctag string
@attribute conn_aftctag string
@attribute conn_befctag_aftctag string
@attribute conn_befctagmsd string
@attribute conn_aftctagmsd string
@attribute abspos_sent integer
@attribute length_par integer
@attribute relpos_sent integer
@attribute mt_num_before integer
```

```
@attribute mt_num_after integer
@attribute mt_total integer
@attribute balance_mt integer
@attribute relpos_mt integer
@attribute mt_rendpres {yes,no}
@attribute mt_rend string
@attribute par_rend string
@attribute pos_def integer
@attribute mt_cap {commonsg,commonpl,propersg,properpl,other}
@attribute mt_art {de,het,een,propersg,properpl,commonsg,commonpl,
  inf,other}
@attribute mt_adj_msd {stell,overtr,vergr,adv,noadj}
@attribute def_art {de,het,een,propersg,properpl,commonsg,commonpl,
  adverb,other}
@attribute def_adj_msd {stell,overtr,vergr,adv,noadj}
@attribute def_dat_postags string
@attribute def_dat {dat,die,om,waarmee,waarbij,waarin,waarvan,voor,
  nodat}
@attribute tfidf integer
@attribute ridf integer
@attribute adridf integer
@attribute bef_bictagmsd string
@attribute bef_bictag string
@attribute man_def {yes,no}

@data
"Art(onbep,zijdofonzijd,neut) N(soort,ev,neut)",
  "Art(onbep,zijdofonzijd, neut) N(soort,ev,neut)
  V(hulpofkopp,ott,3,ev) Art(onbep,zijdofonzijd,neut)
  N(soort,ev,neut) Pron(betr,neut,zelfst) Prep(voor)
  Art(onbep,zijdofonzijd,neut) Adj(attr,stell,onverv)
  N(soort,ev,neut) V(hulpofkopp,ott,3,ev) V(trans,verldw,onverv)
  Punc(punt)","Art(onbep,zijdofonzijd,neut)_N(soort,ev,neut)",
  "Art N","Art N V Art N Pron Prep Art Adj N V V Punc","Art",
  "Art_N","Art_N_Pron","zijn","N","Art","N_Art","N(soort,ev,neut)",
  "Art(onbep,zijdofonzijd,neut)",2,2,1,0,0,0,1,0,no,"empty","li",1,
  commonsg,een,noadj,een,noadj,"Art_N_",dat,5.5077946402,
  -0.0158247555265,-0.0158247555265,yes
"N(soort,mv,neut)","N(soort,mv,neut) V(hulpofkopp,ott,1of2of3,mv)
  Art(bep,zijdofmv,neut) N(soort,mv,neut) Prep(voor)
  Art(onbep,zijdofonzijd,neut) N(soort,ev,neut) Punc(punt)",
  "Art(bep,zijdofmv,neut)","N","N V Art N Prep Art N Punc","Art",
  "Art_N","Art_N_Prep","zijn","N","Art","N_Art","N(soort,mv,neut)",
  "Art(bep,zijdofmv,neut)",1,6,0.166666666666667,2,20,22,
  0.130434782608696,0.818181818181818,no,"empty","p",1,commonpl,
  commonpl,noadj,de,noadj,"nodat",nodat,51.9769439154,
  2.2484848605,10.7833545737,yes
```

D
Bigram properties

This appendix contains a comparison of the most frequent PoS and MSI bigrams in the machine learning datasets. A discussion on how this information has been used can be found in the part on *n*-gram properties in section 4.4.3.

definitions		non-definitions	
%	bigram	%	bigram
8.62	Art_N	8.65	Art_N
7.1	N_Punc	8.12	N_Punc
6.92	N_V	7.11	N_V
6.28	N_Prep	6.37	N_Prep
4.55	Prep_Art	5.63	N_N
4.05	V_Art	4.6	Adj_N
3.93	Adj_N	3.88	Prep_Art
3.52	N_N	3.52	V_Art
3.52	Prep_N	3.27	Prep_N
3.23	V_Punc	2.87	Art_Adj
4.49	Art(onbep,zijdofonzijd,neut)_N(soort,ev,neut)	4.47	N(soort,ev,neut)_Prep(voor)
3.93	N(soort,ev,neut)_Prep(voor)	3.61	N(eigen,ev,neut)_N(eigen,ev,neut)
3.11	V(hulpofkopp,ott,3,ev)_Art(onbep,zijdofonzijd,neut)	3.31	Art(bep,zijdofmv,neut)_N(soort,ev,neut)
1.94	N(eigen,ev,neut)_V(hulpofkopp,ott,3,ev)	2.57	N(soort,ev,neut)_V(hulpofkopp,ott,3,ev)
1.94	N(eigen,ev,neut)_N(eigen,ev,neut)	2.24	Art(onbep,zijdofonzijd,neut)_N(soort,ev,neut)
1.76	N(soort,ev,neut)_V(hulpofkopp,ott,3,ev)	2.02	N(soort,ev,neut)_Punc(punt)
1.61	Prep(voor)_Art(onbep,zijdofonzijd,neut)	1.96	Prep(voor)_Art(bep,zijdofmv,neut)
1.55	Art(bep,zijdofmv,neut)_N(soort,ev,neut)	1.9	Adj(attr,stell,vervneut)_N(soort,ev,neut)
1.55	Prep(voor)_Art(bep,zijdofmv,neut)	1.84	Art(bep,onzijd,neut)_N(soort,ev,neut)
1.53	Prep(voor)_N(soort,ev,neut)	1.58	V(hulpofkopp,ott,3,ev)_Art(onbep,zijdofonzijd,neut)

Table D.1 *Bigrams in* is *patterns*

definitions		non-definitions	
%	bigram	%	bigram
7.36	Art_N	8.69	Art_N
6.79	N_V	7.43	N_Punc
6.79	N_Punc	6.88	N_V
5.38	N_Prep	6.19	N_Prep
5.06	Prep_Art	5.03	Prep_Art
4.06	V_Prep	3.4	Adj_N
3.9	N_N	3.21	N_N
3.55	Prep_N	3	Prep_N
3.49	V_V	2.9	V_Punc
3.24	V_Punc	2.85	V_Prep
3.14	N(soort,ev,neut)_Prep(voor)	3.87	Art(bep,zijdofmv,neut)_N(soort,ev,neut)
2.73	Art(bep,zijdofmv,neut)_N(soort,ev,neut)	3.85	N(soort,ev,neut)_Prep(voor)
2.48	Art(onbep,zijdofonzijd,neut)_N(soort,ev,neut)	2.69	Prep(voor)_Art(bep,zijdofmv,neut)
1.89	Prep(voor)_Art(bep,zijdofmv,neut)	1.95	Art(onbep,zijdofonzijd,neut)_N(soort,ev,neut)
1.63	Prep(voorinf)_V(trans,inf)	1.87	Art(bep,onzijd,neut)_N(soort,ev,neut)
1.63	Prep(voor)_Art(onbep,zijdofonzijd,neut)	1.66	N(soort,mv,neut)_Prep(voor)
1.54	Prep(voor)_Art(bep,onzijd,neut)	1.66	Adj(attr,stell,vervneut)_N(soort,ev,neut)
1.48	N(eigen,ev,neut)_N(eigen,ev,neut)	1.48	N(soort,ev,neut)_Conj(neven)
1.48	Prep(voor)_N(soort,mv,neut)	1.48	N(soort,ev,neut)_Punc(punt)
1.38	Art(bep,onzijd,neut)_N(soort,ev,neut)	1.32	N(soort,ev,neut)_V(trans,ott,3,ev)

Table D.2 *Bigrams in* verb *patterns*

definitions		non-definitions	
%	bigram	%	bigram
11.82	N_Punc	16.73	N_Punc
6.35	Art_N	6.82	Punc_N
5.46	N_Prep	6.16	N_N
4.35	Prep_Art	5.8	Art_N
3.79	N_N	3.5	N_Prep
3.57	N_V	3.35	Adj_N
3.57	Adj_N	3.03	Punc_Punc
3.46	Prep_N	2.94	Prep_Art
3.46	Punc_N	2.28	N_V
2.79	V_Prep	2.21	Prep_N
3.51	N(soort,ev,neut)_Prep(voor)	3.35	N(eigen,ev,neut)_N(eigen,ev,neut)
2.62	N(soort,ev,neut)_Punc(komma)	2.56	Art(bep,zijdofmv,neut)_N(soort,ev,neut)
2.34	Art(bep,zijdofmv,neut)_N(soort,ev,neut)	2.49	N(soort,ev,neut)_Punc(haakopen)
2.01	Prep(voor)_Art(bep,zijdofmv,neut)	2.25	N(soort,ev,neut)_Prep(voor)
1.95	Prep(voor)_N(soort,ev,neut)	1.76	N(soort,ev,neut)_Punc(dubbpunt)
1.95	Art(onbep,zijdofonzijd,neut)_N(soort,ev,neut)	1.68	N(soort,ev,neut)_Punc(haaksluit)
1.78	Adj(attr,stell,vervneut)_N(soort,ev,neut)	1.67	Prep(voor)_Art(bep,zijdofmv,neut)
1.62	Art(bep,onzijd,neut)_N(soort,ev,neut)	1.51	N(soort,ev,neut)_Punc(komma)
1.51	N(eigen,ev,neut)_N(eigen,ev,neut)	1.42	Adj(attr,stell,vervneut)_N(soort,ev,neut)
1.34	Prep(voor)_Art(bep,onzijd,neut)	1.37	N(eigen,ev,neut)_Punc(dubbpunt)

Table D.3 *Bigrams in* punctuation *patterns*

definitions		non-definitions	
%	bigram	%	bigram
7.32	Art_N	8.02	Art_N
6.29	N_V	5.89	N_Punc
5.99	N_Punc	5.13	N_V
5.03	N_Prep	4.86	V_Punc
4.51	Prep_Art	4.85	V_V
4.44	V_Punc	4.78	N_Prep
4.07	V_V	4.07	Prep_Art
3.99	Pron_V	3.7	Adj_N
3.48	Adj_N	3.46	N_Adv
2.88	Prep_N	2.36	Prep_N
3.25	N(soort,ev,neut)_Prep(voor)	3.39	Art(bep,zijdofmv,neut)_N(soort,ev,neut)
3.11	Art(bep,zijdofmv,neut)_N(soort,ev,neut)	3.05	N(soort,ev,neut)_Prep(voor)
2.51	Prep(voor)_Art(bep,zijdofmv,neut)	2.05	Prep(voor)_Art(bep,zijdofmv,neut)
2.14	Art(onbep,zijdofonzijd,neut)_N(soort,ev,neut)	1.95	N(soort,ev,neut)_Adv(gew,geenfunc,stell,onverv)
1.48	N(soort,mv,neut)_Prep(voor)	1.95	Art(onbep,zijdofonzijd,neut)_N(soort,ev,neut)
1.41	N(soort,ev,neut)_Punc(komma)	1.62	Art(bep,onzijd,neut)_N(soort,ev,neut)
1.41	Adj(attr,stell,vervneut)_N(soort,mv,neut)	1.37	Adj(attr,stell,vervneut)_N(soort,ev,neut)
1.33	Prep(voor)_N(soort,ev,neut)	1.32	N(soort,ev,neut)_Punc(komma)
1.26	Prep(voor)_N(soort,mv,neut)	1.24	Punc(komma)_Adv(gew,geenfunc,stell,onverv)
1.26	Punc(komma)_Adv(gew,geenfunc,stell,onverv)	1.22	Prep(voor)_N(soort,ev,neut)

Table D.4 *Bigrams in* pronoun *patterns*

definitions		non-definitions	
%	bigram	%	bigram
7.71	N_Punc	11.11	N_Punc
7.61	Art_N	7.19	Art_N
6.17	N_V	4.56	N_Prep
5.66	N_Prep	4.36	N_N
4.67	Prep_Art	4.3	N_V
3.54	Adj_N	4.16	Punc_N
3.47	N_N	3.64	Prep_Art
3.43	Prep_N	3.63	Adj_N
3.19	V_V	2.86	V_Punc
3.15	V_Prep	2.58	V_V
3.5	N(soort,ev,neut)_Prep(voor)	3.06	Art(bep,zijdofmv,neut)_N(soort,ev,neut)
3.04	Art(onbep,zijdofonzijd,neut)_N(soort,ev,neut)	2.96	N(soort,ev,neut)_Prep(voor)
2.3	Art(bep,zijdofmv,neut)_N(soort,ev,neut)	2.22	N(eigen,ev,neut)_N(eigen,ev,neut)
1.88	Prep(voor)_Art(bep,zijdofmv,neut)	1.93	Prep(voor)_Art(bep,zijdofmv,neut)
1.52	N(eigen,ev,neut)_N(eigen,ev,neut)	1.64	Art(onbep,zijdofonzijd,neut)_N(soort,ev,neut)
1.51	Prep(voor)_N(soort,ev,neut)	1.5	Art(bep,onzijd,neut)_N(soort,ev,neut)
1.43	N(soort,ev,neut)_Punc(komma)	1.49	Adj(attr,stell,vervneut)_N(soort,ev,neut)
1.41	Prep(voor)_Art(bep,onzijd,neut)	1.37	N(soort,ev,neut)_Punc(komma)
1.4	Art(bep,onzijd,neut)_N(soort,ev,neut)	1.23	N(soort,ev,neut)_Punc(haakopen)
1.39	Prep(voor)_Art(onbep,zijdofonzijd,neut)	1.21	Prep(voor)_N(soort,ev,neut)

Table D.5 *Bigrams in all patterns*

E

Connector phrases

This appendix provides an overview of the verbal connector phrases in the definitions and non-definitions extracted with the pattern-based approach (Section 3.7.2). This information has been integrated in the connector properties of the machine learning approach (Section 4.4.3).

Dutch connector	translation	definitions	non-definitions
gebruiken om / voor	use to / for	19.23	11.68
bestaan uit	consist of	10.9	10.66
betekenen	mean	7.69	7.11
noemen	call	7.05	2.03
bevatten	contain	5.13	24.87
omschrijven / definiëren als	describe / define as	5.13	0
staan voor	stands for	4.49	0.51
zijn gericht op	is directed to	3.21	5.08
betreffen	concerns	3.21	4.57
bedoelen	mean	3.21	0.51
omvatten	comprise	2.56	1.52
zorgen voor	take care of	2.56	1.02
verstaan onder	mean by	2.56	0.51
spreken van	speak of	2.56	0
definiëren	define	1.92	3.05
bedoelen om / voor	used to / for	1.92	3.05
heten	call	1.92	2.03
gevormd door	formed by	1.28	1.52
maken mogelijk	make possible	1.28	1.02
mogelijk zijn	is possible	1.28	0.51
staan er	there are	1.28	0
fungeren als	function as	1.28	0
leveren	provides	0.64	10.15
other		7.69	8.58
TOTAL		100	100

Table E.1 *Connector phrases in the* verb *data set*

connector	definitions	non-definitions
colon	67.68	45.92
bracket	32.32	54.08
TOTAL	100	100

Table E.2 *Connector phrases in the* punctuation *data set*

Dutch connector	translation	definitions	non-definitions
waar…	…which	19.48	69.79
hier…	…this	16.88	5.21
dwz	that is	16.88	2.54
zijn	are	27.27	14.97
noemen	call	11.69	0.53
bevatten	contain	3.90	0.27
betekenen	mean	2.60	1.07
bedoelen om / voor	mean to / for	1.30	1.07
other		0	4.55
TOTAL		100	100

Table E.3 *Connector phrases in the* pronoun *data set*

F

Parameter tuning experiments

In this appendix, we list the optimal number of PoS and MSI bigrams and the number of iterations for the bagging experiments described in Chapter 4. The parameter tuning procedure is presented in Section 4.4.7. We based the optimal numbers on the five datasets that have been used.

THE NUMBER OF POS AND MSI BIGRAMS

In this experiment we investigated the optimal number of PoS and PoS+MSI bigrams to be used in the bigram experiments. Results are reported in Table F.1.

THE NUMBER OF ITERATIONS IN THE BAGGING PROCEDURE

In this experiment, different numbers of iterations have been tried to investigate which value should be used for the parameter I. On the basis of the results for the experiments with PoS and MSI bigrams on all datasets, the value of 30 has been used for this parameter. Although there is sometimes a slight improvement observed after this value, these improvements are only marginal and therefore we used 30 iterations. Results of the experiments are provided in Table F.2.

PoS bigrams						PoS+MSI bigrams					
n	all	is	verb	punct	pron	*n*	all	is	verb	punct	pron
10	0.67	0.59	0.61	0.66	0.53	10	0.60	0.69	0.57	0.69	0.49
20	0.70	0.64	0.62	0.75	0.65	20	0.63	0.72	0.53	0.71	0.46
30	0.75	0.66	0.65	0.75	0.66	30	0.67	0.73	0.58	0.71	0.47
40	0.76	0.69	0.63	0.76	0.67	40	0.72	0.75	0.56	0.69	0.56
50	0.76	0.68	0.64	0.75	0.69	50	0.73	0.76	0.59	0.73	0.56
60	0.76	0.67	0.59	0.77	0.69	100	0.75	0.75	0.63	0.75	0.59
70	0.75	0.71	0.60	0.76	0.71	200	0.77	0.80	0.66	0.77	0.62
80	0.76	0.70	0.59	0.75	0.71	300	0.77	0.80	0.65	0.78	0.61
90	0.77	0.69	0.58	0.75	0.70	400	0.79	0.78	0.66	0.78	0.65
100	0.76	0.69	0.58	0.76	0.67	500	0.78	0.78	0.66	0.78	0.64
110	0.77	0.69	0.58	0.73	0.73	600	0.79	0.78	0.66	0.76	0.69
120	0.76	0.69	0.58	0.73	0.73	700	0.79	0.78	0.66	0.77	0.69
all	0.76	0.69	0.58	0.73	0.73	800	0.80	0.78	0.66	0.77	0.69
						900	0.79	0.78	0.66	0.77	0.69
						1000	0.79	0.78	0.66	0.77	0.69
						2000	0.80	0.78	0.66	0.77	0.69
						all	0.79	0.78	0.66	0.77	0.69

Table F.1 *The influence of the number of bigrams n on the AUC score using the Balanced Random Forest classifier*

PoS bigrams						PoS+MSI bigrams				
I	all	is	verb	punct	pron	all	is	verb	punct	pron
0	0.59	0.59	0.59	0.61	0.55	0.59	0.66	0.58	0.57	0.52
5	0.72	0.64	0.59	0.69	0.64	0.75	0.73	0.64	0.74	0.62
10	0.74	0.68	0.62	0.73	0.67	0.77	0.77	0.66	0.75	0.63
15	0.75	0.68	0.63	0.75	0.68	0.78	0.77	0.65	0.77	0.64
20	0.76	0.69	0.63	0.76	0.67	0.79	0.78	0.66	0.78	0.65
25	0.76	0.69	0.64	0.76	0.69	0.79	0.79	0.65	0.79	0.66
30	0.76	0.70	0.64	0.77	0.69	0.80	0.80	0.66	0.79	0.66
40	0.77	0.70	0.64	0.76	0.70	0.80	0.80	0.66	0.79	0.65
50	0.77	0.70	0.64	0.77	0.69	0.80	0.81	0.66	0.79	0.65
60	0.77	0.70	0.65	0.77	0.70	0.80	0.81	0.66	0.80	0.66
70	0.77	0.70	0.65	0.77	0.70	0.81	0.81	0.66	0.80	0.66
80	0.77	0.70	0.65	0.78	0.71	0.81	0.81	0.67	0.80	0.65
90	0.77	0.70	0.66	0.78	0.71	0.81	0.82	0.67	0.80	0.66
100	0.77	0.70	0.65	0.78	0.71	0.81	0.82	0.67	0.80	0.66

Table F.2 *The influence of the number of iterations I in the bagging procedure of the Balanced Random Forest classifier on the AUC score*

G

Results for the individual settings

This appendix presents the results for the sub settings of the linguistic and position settings. The experiments on the basis of this information are discussed in Section 5.2.

	AUC	R	P	F	AUC	R	P	F
		is				**verb**		
grammar		0.86	0.35	0.50		0.78	0.44	0.56
article	0.57	0.46	0.44	0.45	0.43	0.17	0.40	0.24
noun	0.68	0.66	0.49	0.56	0.55	0.40	0.59	0.48
adjective	0.55	0.80	0.38	0.52	0.45	0.70	0.44	0.54
all	0.73	0.67	0.51	0.58	0.61	0.45	0.58	0.51
		punctuation				**pronoun**		
grammar		0.88	0.07	0.12		0.75	0.09	0.16
article	0.58	0.38	0.11	0.17	0.61	0.46	0.17	0.25
noun	0.64	0.65	0.09	0.16	0.64	0.45	0.16	0.24
adjective	0.49	0.21	0.07	0.10	0.47	0.65	0.09	0.16
all	0.63	0.46	0.12	0.19	0.63	0.48	0.17	0.25
		all						
grammar		0.82	0.16	0.26				
article	0.76	0.60	0.33	0.43				
noun	0.79	0.59	0.37	0.45				
adjective	0.73	0.52	0.39	0.45				
all	0.79	0.58	0.36	0.44				

Table G.1 *Results for the linguistic features related to the definiendum*

	AUC	R	P	F	AUC	R	P	F
		is				**verb**		
grammar		0.86	0.35	0.50		0.78	0.44	0.56
article	0.70	0.65	0.55	0.60	0.57	0.46	0.49	0.48
adjective	0.49	0.35	0.34	0.35	0.49	0.69	0.45	0.55
article+adjective	0.70	0.65	0.52	0.58	0.58	0.48	0.53	0.50
relative clause	0.73	0.63	0.60	0.61	0.47	0.64	0.44	0.52
all	0.80	0.62	0.62	0.62	0.55	0.45	0.48	0.47
		punctuation				**pronoun**		
grammar		0.88	0.07	0.12		0.75	0.09	0.16
article	0.56	0.34	0.10	0.16	0.64	0.57	0.15	0.24
adjective	0.47	0.21	0.06	0.09	0.49	0.64	0.10	0.17
article+adjective	0.55	0.29	0.11	0.16	0.63	0.54	0.16	0.24
relative clause	0.59	0.29	0.23	0.25	0.58	0.34	0.21	0.26
all	0.64	0.44	0.16	0.23	0.71	0.48	0.17	0.25
		all						
grammar		0.82	0.16	0.26				
article	0.79	0.58	0.34	0.43				
adjective	0.73	0.52	0.39	0.44				
article+adjective	0.77	0.59	0.34	0.43				
relative clause	0.78	0.58	0.37	0.45				
all	0.80	0.58	0.39	0.47				

Table G.2 *Results for the linguistic features related to the definiens*

	AUC	R	P	F	AUC	R	P	F
		is				**verb**		
grammar		0.86	0.35	0.50		0.78	0.44	0.56
paragraph								
- length paragraph	0.45	0.32	0.29	0.30	0.46	0.35	0.39	0.37
- absolute position	0.61	0.54	0.46	0.50	0.52	0.45	0.47	0.46
- relative position	0.57	0.56	0.45	0.50	0.57	0.44	0.50	0.47
- all	0.63	0.59	0.47	0.52	0.53	0.40	0.51	0.45
position in sentence	0.55	0.83	0.38	0.52	0.55	0.73	0.48	0.57
paragraph + sentence	0.67	0.65	0.49	0.56	0.59	0.42	0.54	0.48
		punctuation				**pronoun**		
grammar		0.88	0.07	0.12		0.75	0.09	0.16
paragraph								
- length paragraph	0.59	0.60	0.09	0.16	0.50	0.28	0.10	0.15
- absolute position	0.57	0.63	0.09	0.15	0.53	0.38	0.11	0.17
- relative position	0.54	0.68	0.08	0.14	0.57	0.38	0.11	0.17
- all	0.59	0.63	0.09	0.15	0.58	0.38	0.15	0.22
position in sentence	0.61	0.58	0.09	0.15	0.68	0.54	0.15	0.24
paragraph + sentence	0.70	0.57	0.12	0.20	0.68	0.45	0.14	0.22
		all						
grammar		0.82	0.16	0.26				
paragraph								
- length paragraph	0.75	0.52	0.35	0.42				
- absolute position	0.75	0.50	0.39	0.44				
- relative position	0.75	0.54	0.34	0.42				
- all	0.75	0.52	0.36	0.42				
position in sentence	0.78	0.61	0.34	0.44				
paragraph + sentence	0.78	0.58	0.33	0.42				

Table G.3 *Results for the features related to the position of the definition*

	AUC	R	P	F	AUC	R	P	F
		is				**verb**		
grammar		0.86	0.35	0.50		0.78	0.44	0.56
absolute	0.63	0.42	0.49	0.46	0.55	0.29	0.46	0.36
absolute + relative	0.62	0.42	0.48	0.45	0.54	0.30	0.46	0.36
absolute + balance	0.61	0.42	0.48	0.45	0.56	0.31	0.48	0.38
all	0.63	0.42	0.48	0.45	0.55	0.30	0.47	0.36
		punctuation				**pronoun**		
grammar		0.88	0.07	0.12		0.75	0.09	0.16
absolute	0.57	0.68	0.08	0.15	0.54	0.59	0.11	0.19
absolute + relative	0.55	0.65	0.08	0.15	0.67	0.53	0.16	0.24
absolute + balance	0.55	0.67	0.08	0.15	0.54	0.61	0.11	0.19
all	0.56	0.67	0.08	0.15	0.67	0.54	0.16	0.25
		all						
grammar		0.82	0.16	0.26				
absolute	0.76	0.52	0.36	0.43				
absolute + relative	0.77	0.59	0.33	0.42				
absolute + balance	0.76	0.51	0.36	0.42				
all	0.77	0.59	0.33	0.42				

Table G.4 *Results for the features related to the position of the definiendum*

H

Machine learning results

This appendix presents the results for the machine learning experiments performed with the BRF classifier using the five types of settings. The description of the experiments can be found in Chapter 5.

	no bigrams				PoS bigrams				PoS+MSI bigrams			
	AUC	R	P	F	AUC	R	P	F	AUC	R	P	F
grammar		0.86	0.35	0.50		0.86	0.35	0.50		0.86	0.35	0.50
bigrams					0.70	0.59	0.49	0.52	0.80	0.59	0.57	0.58
linguistic	0.84	0.67	0.62	0.64	0.84	0.66	0.62	0.64	0.86	0.62	0.65	0.64
connector	0.71	0.66	0.50	0.67	0.74	0.58	0.51	0.54	0.82	0.61	0.66	0.68
position	0.67	0.51	0.46	0.48	0.81	0.63	0.56	0.59	0.83	0.62	0.63	0.62
keywords	0.64	0.46	0.46	0.46	0.77	0.63	0.55	0.59	0.83	0.62	0.62	0.62
layout	0.52	0.17	0.51	0.25	0.68	0.56	0.47	0.51	0.80	0.58	0.58	0.58
ling+conn	0.84	0.66	0.65	0.65	0.83	0.63	0.62	0.62	0.84	0.67	0.64	0.65
ling+pos	0.86	0.70	0.64	0.67	0.89	0.70	0.68	0.69	0.88	0.68	0.69	0.68
ling+lay	0.85	0.66	0.64	0.65	0.84	0.66	0.62	0.64	0.86	0.64	0.66	0.65
ling+kw	0.84	0.68	0.62	0.65	0.85	0.69	0.63	0.66	0.86	0.67	0.68	0.67
conn+pos	0.74	0.55	0.50	0.52	0.80	0.61	0.57	0.59	0.84	0.63	0.62	0.63
conn+lay	0.72	0.64	0.49	0.56	0.77	0.60	0.57	0.58	0.81	0.58	0.63	0.60
conn+kw	0.74	0.60	0.50	0.54	0.78	0.62	0.58	0.60	0.83	0.56	0.62	0.59
pos+lay	0.69	0.52	0.47	0.50	0.79	0.60	0.55	0.57	0.84	0.61	0.64	0.63
pos+kw	0.71	0.59	0.49	0.53	0.81	0.65	0.58	0.61	0.83	0.62	0.64	0.63
kw+lay	0.64	0.48	0.53	0.51	0.77	0.62	0.56	0.59	0.83	0.61	0.64	0.62
conn+ling+pos	0.86	0.71	0.67	0.69	0.87	0.70	0.68	0.69	0.87	0.67	0.67	0.67
conn+ling+lay	0.85	0.65	0.65	0.65	0.84	0.62	0.65	0.64	0.86	0.62	0.65	0.64
conn+ling+kw	0.84	0.67	0.65	0.66	0.86	0.67	0.63	0.65	0.86	0.65	0.66	0.65
conn+pos+lay	0.74	0.56	0.50	0.53	0.78	0.59	0.57	0.58	0.84	0.59	0.62	0.60
conn+pos+kw	0.75	0.59	0.50	0.54	0.80	0.66	0.58	0.61	0.84	0.63	0.61	0.62
conn+lay+kw	0.76	0.59	0.54	0.56	0.80	0.65	0.58	0.61	0.83	0.60	0.63	0.62
ling+pos+lay	0.86	0.70	0.65	0.68	0.89	0.72	0.67	0.69	0.90	0.72	0.67	0.70
ling+pos+kw	0.86	0.70	0.64	0.67	0.88	0.71	0.67	0.69	0.89	0.72	0.69	0.70
ling+lay+kw	0.85	0.68	0.64	0.66	0.86	0.69	0.66	0.68	0.87	0.69	0.66	0.68
pos+lay+kw	0.71	0.59	0.51	0.54	0.81	0.63	0.61	0.62	0.84	0.60	0.64	0.62
conn+ling+pos+lay	0.87	0.70	0.66	0.68	0.88	0.72	0.68	0.70	0.89	0.67	0.71	0.69
conn+ling+pos+kw	0.86	0.70	0.65	0.68	0.88	0.72	0.66	0.68	0.88	0.67	0.66	0.67
conn+ling+lay+kw	0.85	0.66	0.66	0.66	0.85	0.67	0.66	0.66	0.86	0.66	0.66	0.66
conn+pos+lay+kw	0.75	0.60	0.52	0.56	0.80	0.65	0.59	0.62	0.83	0.60	0.62	0.61
ling+pos+lay+kw	0.87	0.72	0.65	0.68	0.88	0.74	0.66	0.70	0.89	0.69	0.71	0.70
all	0.87	0.72	0.65	0.68	0.89	0.72	0.69	0.70	0.89	0.70	0.69	0.70

Table H.1 *Results for is definitions with the BRF classifier*

	no bigrams				PoS bigrams				PoS+MSI bigrams			
	AUC	R	P	F	AUC	R	P	F	AUC	R	P	F
grammar		0.77	0.44	0.56		0.77	0.44	0.56		0.77	0.44	0.56
bigrams					0.64	0.48	0.52	0.50	0.66	0.46	0.57	0.51
connector	0.72	0.54	0.61	0.57	0.72	0.48	0.59	0.53	0.75	0.50	0.62	0.55
position	0.62	0.47	0.53	0.50	0.71	0.51	0.59	0.55	0.74	0.56	0.64	0.59
linguistic	0.63	0.44	0.54	0.49	0.65	0.49	0.54	0.51	0.70	0.48	0.63	0.54
layout	0.52	0.13	0.57	0.21	0.68	0.50	0.58	0.54	0.72	0.49	0.60	0.54
keyword	0.55	0.33	0.51	0.40	0.65	0.48	0.52	0.50	0.64	0.46	0.56	0.50
conn+pos	0.72	0.52	0.60	0.56	0.74	0.50	0.61	0.55	0.75	0.50	0.61	0.55
conn+ling	0.74	0.54	0.63	0.58	0.75	0.50	0.65	0.57	0.75	0.49	0.64	0.56
conn+lay	0.73	0.53	0.61	0.57	0.73	0.49	0.61	0.55	0.76	0.52	0.65	0.57
conn+kw	0.72	0.54	0.62	0.58	0.74	0.53	0.66	0.59	0.72	0.47	0.61	0.53
pos+ling	0.62	0.47	0.51	0.49	0.65	0.48	0.56	0.51	0.72	0.49	0.61	0.54
pos+lay	0.67	0.49	0.57	0.53	0.72	0.53	0.61	0.57	0.74	0.52	0.63	0.57
pos+kw	0.60	0.44	0.50	0.47	0.73	0.54	0.60	0.57	0.72	0.51	0.60	0.55
ling+lay	0.59	0.44	0.49	0.46	0.64	0.48	0.55	0.51	0.71	0.50	0.62	0.55
ling+kw	0.66	0.47	0.54	0.50	0.67	0.52	0.57	0.54	0.72	0.47	0.63	0.54
lay+kw	0.58	0.34	0.54	0.41	0.68	0.49	0.57	0.53	0.69	0.48	0.60	0.53
conn+ling+pos	0.73	0.55	0.61	0.58	0.77	0.57	0.68	0.62	0.75	0.49	0.67	0.57
conn+ling+lay	0.74	0.54	0.64	0.59	0.75	0.53	0.65	0.59	0.75	0.50	0.66	0.57
conn+ling+kw	0.74	0.55	0.63	0.59	0.77	0.53	0.66	0.59	0.78	0.51	0.67	0.58
conn+pos+lay	0.72	0.52	0.60	0.56	0.74	0.49	0.64	0.55	0.77	0.51	0.62	0.56
conn+pos+kw	0.71	0.53	0.60	0.56	0.75	0.49	0.63	0.55	0.76	0.51	0.65	0.57
conn+lay+kw	0.73	0.53	0.63	0.58	0.74	0.49	0.64	0.56	0.76	0.46	0.63	0.53
ling+pos+lay	0.62	0.44	0.50	0.47	0.67	0.48	0.56	0.52	0.71	0.48	0.61	0.54
ling+pos+kw	0.63	0.50	0.53	0.51	0.67	0.48	0.56	0.52	0.75	0.50	0.66	0.57
ling+lay+kw	0.65	0.46	0.54	0.50	0.68	0.51	0.58	0.54	0.70	0.49	0.60	0.54
pos+lay+kw	0.64	0.46	0.53	0.49	0.74	0.56	0.62	0.58	0.71	0.50	0.62	0.55
conn+ling+pos+lay	0.74	0.53	0.62	0.57	0.77	0.52	0.66	0.58	0.77	0.51	0.65	0.57
conn+ling+pos+kw	0.73	0.54	0.63	0.58	0.76	0.51	0.65	0.57	0.78	0.55	0.68	0.61
conn+ling+lay+kw	0.75	0.55	0.64	0.59	0.77	0.53	0.66	0.59	0.77	0.51	0.67	0.58
conn+pos+lay+kw	0.72	0.53	0.60	0.56	0.75	0.50	0.63	0.56	0.78	0.53	0.68	0.60
ling+pos+lay+kw	0.62	0.46	0.50	0.48	0.70	0.51	0.62	0.56	0.70	0.47	0.60	0.53
all	0.73	0.53	0.62	0.57	0.77	0.51	0.66	0.57	0.77	0.51	0.66	0.58

Table H.2 *Results for verb definitions with the BRF classifier*

	no bigrams				PoS bigrams				PoS+MSI bigrams			
	AUC	R	P	F	AUC	R	P	F	AUC	R	P	F
grammar		0.88	0.07	0.12		0.88	0.07	0.12		0.88	0.07	0.12
bigrams					0.77	0.63	0.15	0.24	0.79	0.56	0.18	0.28
position	0.73	0.56	0.14	0.23	0.83	0.66	0.19	0.30	0.82	0.59	0.21	0.31
connector	0.74	0.59	0.13	0.22	0.83	0.62	0.21	0.31	0.81	0.58	0.21	0.30
linguistic	0.72	0.58	0.12	0.20	0.81	0.57	0.18	0.27	0.81	0.49	0.21	0.29
layout	0.64	0.44	0.16	0.23	0.83	0.64	0.19	0.29	0.87	0.63	0.27	0.38
keyword	0.62	0.46	0.09	0.16	0.79	0.63	0.17	0.27	0.80	0.50	0.19	0.28
pos+conn	0.78	0.63	0.16	0.26	0.83	0.63	0.21	0.31	0.86	0.65	0.27	0.38
pos+ling	0.81	0.61	0.17	0.26	0.85	0.63	0.21	0.32	0.84	0.57	0.26	0.35
pos+lay	0.74	0.58	0.16	0.25	0.85	0.66	0.22	0.34	0.84	0.56	0.26	0.36
pos+kw	0.73	0.54	0.15	0.23	0.85	0.69	0.22	0.33	0.83	0.57	0.24	0.33
conn+ling	0.74	0.60	0.15	0.24	0.81	0.64	0.21	0.32	0.81	0.50	0.19	0.28
conn+lay	0.77	0.59	0.16	0.25	0.83	0.66	0.23	0.34	0.86	0.62	0.28	0.38
conn+kw	0.74	0.57	0.15	0.24	0.81	0.62	0.20	0.30	0.82	0.57	0.22	0.32
ling+lay	0.78	0.54	0.16	0.24	0.86	0.67	0.23	0.34	0.86	0.56	0.26	0.36
ling+kw	0.74	0.59	0.14	0.23	0.83	0.60	0.20	0.30	0.82	0.52	0.22	0.31
lay+kw	0.72	0.52	0.14	0.22	0.83	0.61	0.20	0.30	0.86	0.54	0.24	0.34
conn+ling+pos	0.77	0.61	0.16	0.25	0.83	0.62	0.21	0.32	0.83	0.61	0.24	0.35
conn+ling+lay	0.75	0.61	0.16	0.25	0.82	0.65	0.23	0.34	0.85	0.63	0.27	0.37
conn+ling+kw	0.73	0.58	0.15	0.24	0.80	0.63	0.20	0.31	0.84	0.56	0.22	0.32
conn+pos+lay	0.78	0.62	0.16	0.26	0.83	0.66	0.23	0.34	0.88	0.62	0.28	0.38
conn+pos+kw	0.78	0.62	0.16	0.25	0.85	0.64	0.22	0.33	0.85	0.61	0.27	0.37
conn+lay+kw	0.75	0.60	0.16	0.25	0.84	0.63	0.23	0.34	0.86	0.59	0.27	0.37
ling+pos+lay	0.82	0.58	0.18	0.27	0.87	0.66	0.23	0.34	0.88	0.57	0.29	0.39
ling+pos+kw	0.80	0.58	0.16	0.26	0.85	0.65	0.23	0.34	0.84	0.56	0.27	0.37
ling+lay+kw	0.79	0.57	0.16	0.25	0.86	0.62	0.23	0.33	0.86	0.61	0.31	0.41
pos+lay+kw	0.76	0.57	0.16	0.24	0.84	0.62	0.21	0.31	0.86	0.58	0.30	0.40
conn+ling+pos+lay	0.78	0.58	0.16	0.25	0.84	0.65	0.22	0.33	0.86	0.57	0.27	0.37
conn+ling+pos+kw	0.78	0.60	0.16	0.25	0.82	0.64	0.22	0.32	0.84	0.58	0.27	0.37
conn+ling+lay+kw	0.74	0.60	0.16	0.25	0.82	0.60	0.21	0.31	0.85	0.55	0.26	0.35
conn+pos+lay+kw	0.78	0.61	0.16	0.25	0.84	0.63	0.22	0.32	0.85	0.62	0.29	0.40
ling+pos+lay+kw	0.82	0.59	0.18	0.28	0.86	0.63	0.24	0.34	0.86	0.54	0.28	0.37
all	0.78	0.61	0.17	0.27	0.83	0.67	0.23	0.34	0.86	0.63	0.30	0.40

Table H.3 *Results for punctuation definitions with the BRF classifier*

	no bigrams				PoS bigrams				PoS+MSI bigrams			
	AUC	R	P	F	AUC	R	P	F	AUC	R	P	F
grammar		0.73	0.09	0.16		0.73	0.09	0.16		0.73	0.09	0.16
bigrams BRF					0.69	0.42	0.16	0.24	0.66	0.34	0.14	0.20
connector	0.83	0.60	0.24	0.35	0.85	0.54	0.26	0.35	0.81	0.51	0.27	0.35
position	0.70	0.47	0.18	0.27	0.75	0.49	0.20	0.29	0.75	0.48	0.20	0.28
linguistic	0.68	0.45	0.14	0.21	0.74	0.44	0.19	0.26	0.74	0.43	0.20	0.28
keyword	0.59	0.56	0.12	0.20	0.71	0.45	0.18	0.25	0.67	0.36	0.15	0.21
layout	0.42	0.18	0.07	0.10	0.67	0.40	0.15	0.22	0.64	0.38	0.16	0.22
conn+pos	0.82	0.53	0.24	0.33	0.85	0.50	0.26	0.34	0.84	0.54	0.27	0.36
conn+ling	0.82	0.59	0.27	0.37	0.84	0.53	0.28	0.37	0.82	0.50	0.27	0.35
conn+kw	0.83	0.60	0.25	0.35	0.85	0.53	0.29	0.37	0.80	0.43	0.23	0.30
conn+lay	0.83	0.59	0.25	0.35	0.86	0.56	0.28	0.37	0.82	0.49	0.25	0.33
pos+ling	0.72	0.47	0.19	0.27	0.76	0.46	0.21	0.29	0.76	0.46	0.21	0.29
pos+kw	0.70	0.43	0.17	0.24	0.75	0.46	0.20	0.28	0.73	0.47	0.21	0.29
pos+lay	0.71	0.46	0.19	0.27	0.75	0.48	0.20	0.29	0.74	0.44	0.19	0.27
ling+kw	0.70	0.50	0.16	0.25	0.74	0.45	0.20	0.28	0.74	0.40	0.20	0.27
ling+lay	0.70	0.47	0.16	0.24	0.75	0.45	0.20	0.27	0.74	0.41	0.20	0.27
kw+lay	0.57	0.52	0.12	0.19	0.71	0.41	0.16	0.23	0.66	0.34	0.14	0.20
conn+ling+pos	0.82	0.56	0.28	0.37	0.83	0.53	0.27	0.35	0.82	0.48	0.24	0.32
conn+ling+lay	0.84	0.59	0.29	0.39	0.85	0.54	0.29	0.38	0.81	0.47	0.25	0.33
conn+ling+kw	0.83	0.59	0.28	0.38	0.85	0.52	0.29	0.37	0.80	0.47	0.26	0.34
conn+pos+lay	0.82	0.53	0.25	0.34	0.84	0.55	0.27	0.37	0.79	0.49	0.25	0.33
conn+pos+kw	0.82	0.54	0.25	0.34	0.83	0.57	0.27	0.36	0.81	0.47	0.24	0.32
conn+lay+kw	0.84	0.60	0.26	0.36	0.84	0.51	0.28	0.36	0.81	0.50	0.25	0.34
ling+pos+lay	0.72	0.47	0.20	0.28	0.76	0.46	0.22	0.30	0.76	0.44	0.21	0.29
ling+pos+kw	0.72	0.46	0.19	0.27	0.77	0.47	0.22	0.30	0.76	0.46	0.21	0.29
ling+lay+kw	0.72	0.49	0.17	0.26	0.74	0.44	0.20	0.28	0.72	0.38	0.19	0.25
pos+lay+kw	0.70	0.46	0.19	0.27	0.75	0.47	0.22	0.30	0.76	0.45	0.21	0.29
conn+ling+pos+lay	0.83	0.58	0.29	0.38	0.83	0.53	0.28	0.37	0.82	0.52	0.25	0.34
conn+ling+pos+kw	0.82	0.57	0.28	0.38	0.83	0.54	0.28	0.37	0.81	0.48	0.25	0.33
conn+ling+lay+kw	0.84	0.59	0.29	0.39	0.85	0.55	0.30	0.39	0.82	0.51	0.28	0.36
conn+pos+lay+kw	0.82	0.54	0.26	0.35	0.82	0.52	0.26	0.35	0.81	0.52	0.24	0.33
ling+pos+lay+kw	0.73	0.46	0.20	0.28	0.75	0.43	0.21	0.28	0.76	0.45	0.23	0.31
all	0.84	0.60	0.29	0.39	0.84	0.55	0.29	0.38	0.79	0.47	0.25	0.33

Table H.4 *Results for pronoun definitions with the BRF classifier*

	no bigrams				PoS bigrams				PoS+MSI bigrams			
	AUC	R	P	F	AUC	R	P	F	AUC	R	P	F
grammar		0.81	0.16	0.26		0.81	0.16	0.26		0.81	0.16	0.26
bigrams BRF					0.81	0.60	0.35	0.44	0.82	0.56	0.38	0.45
connector	0.84	0.62	0.37	0.46	0.87	0.63	0.44	0.51	0.87	0.60	0.45	0.51
linguistic	0.82	0.60	0.37	0.46	0.86	0.61	0.41	0.49	0.85	0.60	0.43	0.50
position	0.80	0.59	0.34	0.43	0.86	0.61	0.41	0.49	0.86	0.62	0.44	0.52
keyword	0.78	0.56	0.35	0.43	0.83	0.58	0.38	0.46	0.83	0.59	0.39	0.47
layout	0.77	0.54	0.37	0.44	0.83	0.60	0.38	0.46	0.84	0.59	0.39	0.47
conn+ling	0.85	0.64	0.40	0.49	0.88	0.63	0.45	0.53	0.87	0.59	0.47	0.52
conn+pos	0.84	0.62	0.38	0.47	0.87	0.63	0.43	0.51	0.88	0.62	0.45	0.52
conn+lay	0.84	0.63	0.38	0.47	0.87	0.62	0.44	0.51	0.87	0.61	0.46	0.52
conn+kw	0.84	0.63	0.37	0.47	0.88	0.63	0.43	0.51	0.87	0.60	0.46	0.52
ling+pos	0.86	0.61	0.41	0.49	0.88	0.64	0.46	0.54	0.88	0.61	0.46	0.53
ling+lay	0.83	0.62	0.37	0.47	0.87	0.61	0.43	0.51	0.86	0.60	0.44	0.51
ling+kw	0.83	0.61	0.37	0.46	0.87	0.61	0.43	0.51	0.86	0.59	0.45	0.51
pos+lay	0.81	0.59	0.38	0.46	0.86	0.62	0.42	0.50	0.87	0.63	0.44	0.52
pos+kw	0.81	0.59	0.38	0.47	0.86	0.62	0.43	0.51	0.86	0.62	0.44	0.51
lay+kw	0.79	0.57	0.37	0.45	0.85	0.60	0.41	0.49	0.85	0.61	0.41	0.49
conn+ling+pos	0.86	0.65	0.42	0.51	0.89	0.64	0.46	0.54	0.89	0.61	0.48	0.53
conn+ling+lay	0.86	0.64	0.41	0.50	0.88	0.63	0.46	0.53	0.88	0.60	0.48	0.53
conn+ling+kw	0.85	0.63	0.41	0.50	0.88	0.64	0.46	0.53	0.88	0.60	0.48	0.53
conn+pos+lay	0.84	0.62	0.39	0.48	0.88	0.63	0.44	0.52	0.89	0.63	0.48	0.54
conn+pos+kw	0.84	0.63	0.38	0.47	0.88	0.65	0.44	0.53	0.88	0.62	0.47	0.53
conn+lay+kw	0.84	0.64	0.39	0.48	0.88	0.64	0.45	0.53	0.88	0.60	0.46	0.52
ling+pos+lay	0.86	0.62	0.42	0.50	0.89	0.65	0.47	0.55	0.88	0.61	0.48	0.54
ling+pos+kw	0.86	0.61	0.41	0.49	0.89	0.62	0.46	0.53	0.87	0.61	0.46	0.52
ling+lay+kw	0.84	0.61	0.38	0.47	0.88	0.61	0.45	0.52	0.87	0.61	0.45	0.52
pos+lay+kw	0.82	0.61	0.38	0.47	0.87	0.62	0.45	0.52	0.87	0.61	0.44	0.51
conn+ling+pos+lay	0.87	0.64	0.43	0.51	0.89	0.64	0.47	0.54	0.89	0.61	0.47	0.54
conn+ling+pos+kw	0.87	0.65	0.42	0.51	0.89	0.64	0.47	0.54	0.89	0.61	0.48	0.54
conn+ling+lay+kw	0.86	0.63	0.42	0.51	0.89	0.64	0.48	0.55	0.88	0.59	0.48	0.53
conn+pos+lay+kw	0.84	0.62	0.39	0.48	0.88	0.65	0.46	0.54	0.88	0.62	0.48	0.54
ling+pos+lay+kw	0.86	0.62	0.43	0.51	0.89	0.64	0.47	0.54	0.88	0.62	0.48	0.54
all	0.87	0.64	0.44	0.52	0.89	0.64	0.48	0.55	0.89	0.63	0.50	0.56

Table H.5 *Results for all definitions with the BRF classifier*

Bibliography

I. Androutsopoulos and D. Galanis. A practically unsupervised learning method to identify single-snippet answers to definition questions on the web. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP)*, pages 323–330, 2005.

Aristotle. *Posterior Analytics - book II*. 71a. Translated by J. Barnes (2007). Oxford Clarendon Press.

Aristotle. *Topics - book I*. 100a. Translated by R. Smith (1997). Oxford University Press.

G. Barnbrook. *Defining language: a local grammar of definition sentences*. John Benjamines Publishing Company, 2002.

R. Bekkerman and J. Allan. Using bigrams in text categorization. Technical report, Technical Report IR, 2003.

J. Berghmans. Wotan - een probabilistische grammatikale tagger voor het Nederlands. Master's thesis, TOSCA Research Group, University of Nijmegen, Nijmegen, The Netherlands, 1995.

S. Blair-Goldensohn, K. R. McKeown, and A. Hazen Schlaikjer. *New directions in Question Answering*, chapter Answering definitional questions: A hybrid approach. AAAI Press, 2004.

C. Borg, M. Rosner, and G. Pace. Evolutionary algorithms for definition extraction. In *Proceedings of the Workshop Definition Extraction (wDE) at RANLP 2009*, 2009.

R. Borsodi. *The definition of definition*. Porter Sargent Publisher, 1967.

A. van den Bosch and W. Daelemans. Memory-based morphological analysis. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*, pages 285–292, 1999.

A.P. Bradley. The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Pattern Recognition*, 30(7):1145–1159, 1997.

L. Breiman. Random Forests. *Machine Learning*, 45(1):5–32, 2001.

L. Breiman. Bagging predictors. *Machine Learning*, 24(2):123–140, 1996.

L. Breiman, J.H. Friedman, R.A. Olshen, C.J. Stone, L. Breiman, J.H. Freidman, R.A. Olshen, and C.J. Stone. *Classification and regression trees*. Wadsworth, 1984.

P.W. Bridgman. *The logic of modern physics*. Macmillan, 1928.

L. Carroll. *The annotated Alice. Alice's adventures in wonderland & through the looking-glass*. W.W. Norton & Company, Inc., 1999.

O. Chapelle, B. Schölkopf, and A. Zien. *Semi-supervised learning*. MIT Press, 2006.

E. Charniak. A maximum-entropy-inspired parser. In *Proceedings of NAACL-2000*, 2000.

N.V. Chawla. Data mining for imbalanced datasets: An overview. In O.Z. Maimon and L. Rokach, editors, *Data Mining and Knowledge Discovery Handbook*, pages 853–867. Springer, 2005.

N.V. Chawla, K.W. Bowyer, L.O. Hall, and W.P. Kegelmeyer. SMOTE: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16(1):321–357, 2002.

C. Chen, A. Liaw, and L. Breiman. Using Random Forest to learn imbalanced data. Technical Report 666, University of California, Berkeley, 2004. URL `http://www.stat.berkeley.edu/tech-reports/666.pdf`.

K.W. Church and W.A. Gale. Poisson mixtures. *Natural Language Engineering*, 1(2):163–190, 1995a.

K.W. Church and W.A. Gale. Inverse Document Frequency (IDF): A measure of deviations from Poisson. In *Proceedings of the Third Workshop on Very Large Corpora*, pages 121–130, 1995b.

W.W. Cohen. Fast effective rule induction. In *Machine Learning: Proceedings of the 12th internation conference*, pages 115–123, 1995.

P.A. Coppen, W. Haeseryn, and F. de Vriend. E-ANS, 2002. URL `http://www.let.ru.nl/ans/e-ans/index.html`.

H. Cui, M.Y. Kan, and T.S. Chua. Soft pattern matching models for definitional question answering. *ACM Transactions on Information Systems (TOIS)*, 25(2):8, 2007.

W. Daelemans, J. Zavrel, and P. Berck. Part-of-speech tagging for Dutch with MBT, a memory-based tagger generator. In *Informatiewetenschap*, pages 33–40, 1996a.

W. Daelemans, J. Zavrel, P. Berck, and S. Gillis. MBT: A memory-based part of speech tagger generator. In E. Ejerhed and I. Dagan, editors, *Proceedings of the Fourth Workshop on Very Large Corpora*, pages 14–27, 1996b.

W. Daelemans, J. Zavrel, K. van der Sloot, and A. van den Bosch. Timbl: Tilburg memory-based learner. 2007.

Ł. Degórski, M. Marcińczuk, and A. Przepiórkowski. Definition extraction using a sequential combination of baseline grammars and machine learning classifiers. In *Proceedings of LREC 2008*, 2008.

R. Del Gaudio and A. Branco. Automatic extraction of definitions in Portuguese: A rule-based approach. In *Proceeding of 2nd Workshop on Text Mining and Applications at EPIA 2007*, 2007.

R. Del Gaudio and A. Branco. Extraction of definitions in Portuguese: An imbalanced data set problem. In *Proceedings of Text Mining and Applications at EPIA 2009*, 2009.

F. van Eynde, J. Zavrel, and W. Daelemans. Part of Speech tagging and lemmatisation for the Spoken Dutch Corpus. In *Proceedings of LREC 2000*, pages 1427–1433, 2000.

I. Fahmi and G. Bouma. Learning to identify definitions using syntactic features. In R. Basili and A. Moschitti, editors, *Proceedings of the EACL workshop on Learning Structured Information in Natural Language Applications*, 2006.

T. Fawcett. ROC graphs: Notes and practical considerations for researchers. *Machine Learning*, 31, 2004.

C. Fellbaum, editor. *WordNet – An electronic lexical database*. MIT Press, 1998.

E. Frank, G.W. Paynter, I.H. Witten, C. Gutwin, and C.G. Nevill-Manning. Domain-specific keyphrase extraction. In *Internation Joint Conference on Artificial Intelligence*, volume 16, pages 668–673. Lawrence Erlbaum Associates LTD, 1999.

J.D. Gergonne. Essai sur la theorie des definitions. In *Annales de mathématiques pures et appliquées*, volume 9, 1818.

A. Gupta. Definitions, 2008. URL `http://plato.stanford.edu/entries/definitions/`.

J. Han and M. Kamber. *Data mining: concepts and techniques*. Morgan Kaufman, 2006.

K.S. Han, Y.I. Song, K. Sang-Bum, and H.C. Rim. Answer extraction and ranking strategies for definitional question answering using linguistic features and definition terminology. *Information Processing and Management*, 45:353–364, 2007.

M. Hearst. Automatic acquisition of hyponyms from large text corpora. In *Proceedings of COLING 92*, pages 539–545, 1992.

I. Hendrickx, G. Bouma, F. Coppens, W. Daelemans, V. Hoste, G. Kloosterman, A. Mineur, J. Van Der Vloet, and J.L. Verschelde. Coreference resolution for extracting answers for Dutch. In *Proceedings of LREC 2008*, Marrakech, Morocco, 2008.

V. Hoste, I. Hendrickx, and W. Daelemans. Disambiguation of the neuter pronoun and its effect on pronominal coreference resolution. *Lecture Notes in Artificial Intelligence*, 4629:48–55, 2007. Proceedings of the 10th International Conference on Text, Speech and Dialogue, Plzen, Czech Republic.

S. Hunston and J. Sinclair. A local grammar of evaluation. *Evaluation in Text: Authorial stance and the construction of discourse*, pages 74–101, 2000.

N. Ide and K. Suderman. XML Corpus Encoding Standard, document XCES 0.2. Technical report, Department of Computer Science, Vassar College, and Equipe Langue et Dialogue, New York, USA and LORIA/CNRS, Vandoeuvre-lés-Nancy, France,, 2002. `http://www.xces.org/`.

A. Iftene, D. Trandabăţ, and I. Pistol. Grammar-based automatic extraction of definitions. applications for romanian. In *Workshop proceedings RANLP 2007*, 2007.

W.E. Johnson. *Logic*. The University Press, 1921.

H. Joho and M. Sanderson. Large scale testing of a descriptive phrase finder. In *Proceedings of the 1st Human Language Technology Conference*, pages 219–221, 2001.

H. Joho and M. Sanderson. Retrieving descriptive phrases from large amounts of free text. In *Proceedings of the ninth international conference on Information and knowledge management*, pages 180–186. ACM Press New York, NY, USA, 2000.

G. Kennedy. *An introduction to corpus linguistics*. Addison Wesley Longman Limited, 1998.

J. J. Klavans and S. Muresan. Definder: Rule-based methods for the extraction of medical terminology and their associated definitions from on-line text. *American Medical Informatics Association*, 2000.

Ł. Kobyliński and A. Przepiórkowski. Definition extraction with Balanced Random Forests. In B. Nordström and A. Ranta, editors, *Advances in Natural Language Processing: Proceedings of the 6th International Conference on Natural Language Processing, GoTAL 2008*, pages 237–247. Springer Verlag, LNAI series 5221, 2008.

S.B. Kotsiantis. Supervised machine learning: A review of classification techniques. *Informatica*, 31:249–268, 2007.

M. Kubat, R. Holte, and S. Matwin. Machine learning for the detection of oil spills in satellite radar images. *Machine Learning*, 30:195–215, 1998.

D. Laertius. The lives and opinions of eminent philosophers. URL `http://classicpersuasion.org/pw/diogenes/dldiogenes.htm`. Translated by C.D. Yonge (1853).

L. Lemnitzer and P. Monachesi. Extraction and evaluation of keywords from learning objects – a multilingual approach. In *Proceedings of LREC 2008*, 2008.

L. Lemnitzer, C. Vertan, A. Killing, K. Simov, D. Evans, D. Cristea, and P. Monachesi. Improving the search for learning objects with keywords and ontologies. In *Proceedings of ECTEL 2007*. Springer Verlag, 2007.

C.I. Lewis. *Mind and the world-order*. Charles Scribner's Sons Chicago, 1929.

R. Likert. A technique for the measurement of attitudes. *Archives of Psychology*, 140:1–55, 1932.

C.X. Ling, J. Huang, and H. Zhang. AUC: a statistically consistent and more discriminating measure than accuracy. In *International Joint Conference on Artificial Intelligence*, volume 18, pages 519–526. Lawrence Erlbaum Associates Ltd, 2003.

J. Locke. *An essay concerning human understanding*. 1690. Translated by P.H. Nidditch (1975). Oxford University Press.

C.E. Metz. Basic principles of ROC analysis. *Seminars in nuclear medicine*, 8(4):283–298, 1978.

S. Miliaraki and I. Androutsopoulos. Learning to identify single-snippet answers to definition questions. In *Proceedings of COLING 2004*, pages 1360–1366, 2004.

T. M. Mitchell. *Machine learning*. McGraw-Hill, 1997.

P. Monachesi, L. Lemnitzer, and K. Simov. Language Technology for eLearning. In W. Nejdl and K. Tochtermann, editors, *Proceedings of EC-TEL 2006*, pages 667–672. Springer LNCS, 2006.

S. Muresan and J. Klavans. A method for automatically building and evaluating dictionary resources. In *Proceedings of the Language Resources and Evaluation Conference*, 2002.

W.T. Parry and E.A. Hacker. *Aristotelian Logic*. SUNY Press, 1991.

S.C. Pepper. *The Basis of Criticism in the Arts*. Harvard University Press, 1945.

Plato. *Republic - Book VII*. 514a-541b. Translated by R.E. Allen (2006). Yale University Press.

J. Prager, D. Radev, and K. Czuba. Answering what-is questions by virtual annotation. In *Proceedings of the 1st Human Language Technology Conference*, pages 26–30, 2001.

J. Prager, J. Chu-Carroll, and K. Czuba. Use of WordNet hypernyms for answering What-Is-questions. In *Proceedings of TREC-2001*, 2002.

A. Przepiórkowski, Ł. Degórski, M. Spousta, K. Simov, P. Osenova, L. Lemnitzer, V. Kubon, and B. Wójtowicz. Towards the automatic extraction of definitions in Slavic. In *Proceedings of BSNLP workshop at ACL*, 2007a.

A. Przepiórkowski, Ł. Degórski, and B. Wójtowicz. Towards the automatic extraction of definitions in Polish. In *Proceedings of LTC 2007*, 2007b.

J. Rebeyrolle and L. Tanguy. Repérage automatique de structures linguistiques en corpus: le cas des énoncés définitoires. *Cahiers de Grammaire*, 25:153–174, 2001.

J. Renkema. *Schrijfwijzer: Handboek voor duidelijk taalgebruik*. Sdu Uitgevers, Den Haag, 2002.

C.J. van Rijsbergen. *Information Retrieval*. Butterworth, 1979.

R. Robinson. *Definitions*. Oxford University Press, 1972.

H. Saggion. Identifying definitions in text collections for question answering. In *Proceedings of the Language Resources and Evaluation Conference*, 2004.

G. Salton and C. Buckley. Term weighting approaches in automatic text retrieval. *Information Processing and Management*, 24(5):513–523, 1988.

R.E. Schapire and Y. Singer. BoosTexter: A boosting-based system for text categorization. *Machine Learning*, 39(2):135–168, 2000.

F. Sclano and P. Velardi. Termextractor: a web application to learn the shared terminology of emergent web communities. In *Proceedings of the 3rd International Conference on Interoperability for Enterprise Software and Applications (I-ESA)*, pages 28–30. Springer, 2007.

G. Sierra, R. Alarcon, C. Aguilar, and C. Bach. Definitional verbal patterns for semantic relation extraction. *Terminology*, 14(1):74–98, 2008.

A. Singhal, G. Salton, and C. Buckley. Length normalization in degraded text collections. In *Fifth Annual Symposium on Document Analysis and Information Retrieval*, pages 149–162. Cornell University, 1995.

A. Singhal, C. Buckley, and M. Mitra. Pivoted document length normalization. In *Proceedings of the Nineteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 21–29, 1996.

B. de Spinoza. *Ethics*. 1677. Translated by G.H.R. Parkinson (2000). Oxford University Press.

A. Storrer and S. Wellinghof. Automated detection and annotation of term definitions in German text corpora. In *Proceedings of LREC 2006*, 2006.

C.M. Tan, Y.F. Wang, and C.D. Lee. The use of bigrams to enhance text categorization. *Information Processing and Management*, 38(4):529–546, 2002.

H. Tanev. Socrates – a Question Answering prototype for Bulgarian. In *Proceedings of RANLP*, 2004.

R. Tobin. Lxtransduce, a replacement for fsgmatch, 2005. `http://www.ltg.ed.ac.uk/~richard/ltxml2/lxtransduce-manual.html`.

P. Velardi, R. Navigli, and P. D'Amadio. Mining the web to create specialized glossaries. *IEEE Intelligent Systems*, 23(5):18–25, 2008.

E.M. Voorhees. The TREC question answering track. *Natural Language Engineering archive*, 7(4):361–378, 2002.

S. Walter and M. Pinkal. Automatic extraction of definitions from German court decisions. In *Proceedings of the workshop on information extraction beyond the document*, pages 20–28, 2006.

G.M. Weiss. Mining with rarity: a unifying framework. *ACM SIGKDD Explorations Newsletter*, 6(1):7–19, 2004.

E.N. Westerhout. Extraction of definitions using grammar-enhanced machine learning. In *Proceedings of the Student Research Workshop at EACL 2009*, pages 88–96, Athens, Greece, 2009a. Association for Computational Linguistics.

E.N. Westerhout. Definition extraction using linguistic and structural features. In *Proceedings of the Workshop Definition Extraction (wDE) at RANLP 2009*, 2009b.

E.N. Westerhout and P. Monachesi. Extraction of Dutch definitory contexts for elearning purposes. In *Proceedings of CLIN 2006*, 2007a.

E.N. Westerhout and P. Monachesi. Combining pattern-based and machine learning methods to detect definitions for elearning purposes. In *Proceedings of RANLP 2007 Workshop "Natural Language Processing and Knowledge Representation for eLearning Environments"*, 2007b.

E.N. Westerhout and P. Monachesi. Creating glossaries using pattern-based and machine learning techniques. In *Proceedings of LREC 2008*, 2008.

I. Witten and E. Frank. *Data mining: Practical machine learning tools and techniques*. Morgan Kaufman Publishers, 2005.

I.H. Witten, G.W. Paynter, E. Frank, C. Gutwin, C.G. Nevill-Manning, and N.Z. Hamilton. KEA: Practical automatic keyphrase extraction. pages 254–256, 1999.

Lei Yu and Huan Liu. Efficient feature selection via analysis of relevance and redundancy. *Journal of Machine Learning Research*, 5:1205–1224, 2004.

Samenvatting

In deze dissertatie staat het automatisch extraheren van definities centraal. Een definitie is een beschrijving van de betekenis van een term. Het automatisch identificeren van definities is een relevante taak voor verschillende toepassingen. De context waarop deze dissertatie zich richt, is de semi-automatische ontwikkeling van verklarende woordenlijsten, ook wel glossariums genoemd. Een glossarium is een lijst met definities voor de belangrijkste termen die in een tekst worden besproken. Het gebruik van een glossarium helpt bij het bestuderen van een tekst, omdat het de hoofdtermen van een document op een rijtje zet en de betekenis hiervan overzichtelijk aanbiedt.

Omdat er niet voor elke tekst een glossarium beschikbaar is, moeten leerders vaak zelf op zoek naar definities. Leerders moeten hiervoor in staat zijn om uit de verschillende mogelijke definities de beste te selecteren, dat is, een definitie die correct is en aansluit bij het domein van de leertekst. Een bijkomend probleem zijn nieuwe termen, waarvoor nog geen definities voorhanden zijn. Leerders zouden erg geholpen zijn wanneer een glossarium automatisch gemaakt kan worden op basis van de leertekst. In dit onderzoek presenteren wij een methode om deze taak semi-automatisch uit te voeren. Het ontwikkelen van deze methode is gestart binnen het Europese project 'Language Technology for eLearning' (LT4eL), waarin taaltechnologische toepassingen zijn ontwikkeld voor de ondersteuning van eLearningactiviteiten.

Voordat er een methode ontwikkeld kan worden, is het allereerst noodzakelijk om te definiëren wat verstaan wordt onder definities. Een relevant onderscheid is hierbij het verschil tussen definities in enge en ruime zin. Een definitie in enge zin geeft de exacte betekenis van een term, wat bijvoorbeeld belangrijk is in een woordenboek. In een definitie in ruime zin is het voldoende wanneer de definitie een algemene beschrijving geeft, bijvoorbeeld door aan te geven tot welke categorie een term behoort. Het onderscheid tussen globale en exacte definities is gradueel.

Voor glossariums zijn definities in enge en ruime zin allebei nuttig. Om definities te kunnen onderscheiden van niet-definities is in hoofdstuk 2 onderzocht wat voor indelingen er zijn voor het classificeren van definities. De meest gangbare methode is gebaseerd op semantische ei-

genschappen van definities en deelt definities in op basis van de manier die gebruikt is om de term te verduidelijken (e.g. gebruik van synoniemen, relationele definities, contextuele definities). Omdat semantische eigenschappen moeilijk automatisch zijn te identificeren, wordt in deze studie een patroongebaseerde methode toegepast.

In deze benadering wordt ervan uitgegaan dat een definitie bestaat uit minimaal drie onderdelen: het te definiëren begrip ('definiendum'), de beschrijving van dit begrip ('definiens') en een frase die het definiendum en de definiens met elkaar verbindt ('connector'). Op basis van de verschillende typen connectors die gebruikt zijn in een collectie van 600 handmatig geselecteerde definities, onderscheidt de patroongebaseerde methode vier typen definities. Dit zijn de *is*, *werkwoord*, *interpunctie* en *voornaamwoord* definities, zie de voorbeelden in 16:

(16) a. ***is* definitie**: [Gnuplot]$_{DEFINIENDUM}$ [is]$_{CONN}$ [een programma om grafieken te maken.]$_{DEFINIENS}$

b. ***werkwoord* definitie**: [eLearning]$_{DEFINIENDUM}$ [omvat]$_{CONN}$ [hulpmiddelen en toepassingen die via het internet beschikbaar zijn en creatieve mogelijkheden bieden om de leerervaring te verbeteren.]$_{DEFINIENS}$

c. ***interpunctie* definitie**: [Passen]$_{DEFINIENDUM}$ [:]$_{CONN}$ [plastic kaarten voorzien van een magnetische strip, die door een gleuf gehaald worden, waardoor de gebruiker zich kan identificeren en toegang krijgt tot bepaalde faciliteiten.]$_{DEFINIENS}$

d. ***voornaamwoord* definitie**: [Dedicated readers]$_{DEFINIENDUM}$. [Dit zijn]$_{CONN}$ [speciale apparaten, ontwikkeld met het exclusieve doel e-boeken te kunnen lezen.]$_{DEFINIENS}$

De lexico-syntactische patronen van het definiendum en het begin van de definiens blijken ook beperkt te zijn. Zoals in de voorbeelden van 16 te zien is, is het definiendum over het algemeen een naamwoordgroep terwijl de definiens hier vaak ook mee begint. Naast deze

frequent gebruikte patronen bevatten de definities ook minder gangbare patronen (e.g. het gebruik van adverbiumgroepen, het gebruik van apposities). De connectorfrasen en de lexico-syntactische patronen zijn geïntegreerd in vier grammatica's die gebruikt zijn om de vier typen definities te onderscheiden van niet-definities (Zie hoofdstuk 3). Met deze grammatica's blijkt het mogelijk het merendeel van de definities te vinden. Een nadeel van de patroongebaseerde methode is het feit dat de connectorfrasen ook vaak gebruikt worden in niet-definities. Dit is met name een probleem voor de *interpunctie* en *voornaamwoord* definities.

Om dit probleem op te lossen is de patroongebaseerde methode aangevuld met een tweede stap, waarin de zinnen die geen definities zijn zoveel mogelijk uit de datasets worden gefilterd. Er is hiervoor in hoofdstuk 4 per definitietype onderzocht voor een aantal tekstkenmerken of zij zich verschillend gedragen in definities en niet-definities:

1. **Connectoreigenschappen**: het aantal incorrect geëxtraheerde zinnen blijkt samen te hangen met de gebruikte connectorfrase. Verder kijken we naar de linguïstische categorie van de woorden direct links en rechts van de connectorfrase.

2. **Linguïstische eigenschappen van definiendum en definiens**: in het definiendum is gekeken naar het type lidwoord (de, het, een), het type adjectief (stellend, vergrotend, overtreffend) en het type zelfstandig naamwoord (eigennaam of soortnaam, enkelvoud of meervoud). Voor de definiens is gekeken naar het lidwoord en adjectief van de eerste naamwoordgroep en naar de eventuele aanwezigheid van een bijvoeglijke bijzin. Er blijken aanzienlijke verschillen te bestaan voor deze kenmerken, met name in de *is* definities. De verschillen zijn het kleinst in de *werkwoord* definities.

3. **Positionele eigenschappen**: definities blijken relatief meer aan het begin van een paragraaf voor te komen in vergelijking met niet-definities, met name bij de *is* definities. Een tweede positioneel aspect dat onderzocht is, betreft de positie van de definiendum binnen de tekst. Het definiendum wordt relatief meer

gebruikt nadat het gedefinieerd is dan ervoor. Dit geldt voor alle typen definities.

4. **Lay-outeigenschappen**: met betrekking tot de lay-out van de definitie valt op dat *interpunctie* definities relatief vaak gebruikt worden in lijsten. De overige definities worden met name binnen paragrafen gebruikt. Onderzoek van de lay-outeigenschappen van de definiens heeft aangetoond dat in definities vaker specifieke lay-outeigenschappen worden gebruikt dan in niet-definities.

5. **Trefwoordeigenschappen**: voor de *is* definities duidt een hogere trefwoordscore erop dat het waarschijnlijker is dat een zin een definities is. Bij de andere typen is dit verschil minder aanwezig.

Op basis van de vijf eigenschappen zijn machine learning classifiers getraind (hoofdstuk 5). De connector, linguïstische, en positionele eigenschappen blijken over het algemeen het meest relevant voor de classificatie, terwijl de lay-out- en trefwoordeigenschappen individueel het minst geschikt zijn voor het succesvol classificeren van definities. Wanneer verschillende soorten eigenschappen gecombineerd worden, blijkt echter dat deze eigenschappen wel relevante informatie toevoegen. Met name bij de *interpunctie* definities draagt het gebruik van lay-outkenmerken in hoge mate bij aan de classificatie.

Naast de vijf genoemde eigenschappen is er ook gekeken naar het nut van linguïstische bigrammen voor de classificatie. Er is een onderscheid gemaakt tussen part-of-speech tag bigrammen en morpho-syntactische bigrammen. Wanneer alleen bigrammen worden gebruikt, zijn de classificatieresultaten vaak vergelijkbaar of minder goed dan wanneer de beste individuele settings worden gebruikt. Gebruik van alleen de bigrammen biedt echter een goede basis die verder verbeterd kan worden door de individuele settings eraan toe te toevoegen. Welke informatie de meeste toegevoegde waarde biedt, hangt deels van het definitietype af. Over het algemeen zijn de connector, linguïstische en positionele eigenschappen het meest relevant. Het type bigrammen dat gebruikt wordt, maakt vaak niet uit wanneer er meerdere eigenschappen worden toegevoegd; de algemene part-of-speech tag bigrammen presteren in dit geval vaak vergelijkbaar met de morpho-syntactische

bigrammen. Wanneer er slechts één of twee eigenschappen worden gebruikt naast de bigrammen, is het meestal beter om de meer gedetailleerde morpho-syntactische bigrammen te gebruiken.

In hoofdstuk 6 wordt de dissertatie afgesloten met conclusies, een discussie van de ontwikkelde methode, en enkele suggesties voor toekomstig onderzoek op het gebied van definitie-extractie. De resultaten laten zien dat de hier voorgestelde methode, waarin patronen voor definities en machine learning technieken voor de bovengenoemde eigenschappen gecombineerd worden, gelijkwaardige of betere resultaten oplevert dan andere benaderingen. Verder kan geconcludeerd worden dat de diversiteit van patronen die met onze methode wordt geïdentificeerd groot is in vergelijking met de meeste bestaande methoden. Het onderscheid in de vier typen definities dat gemaakt werd in de patroongebaseerde benadering is niet zinvol wanneer machine learning technieken worden gebruikt. In dit geval is het beter om één dataset voor alle typen te gebruiken.

Concluderend, met de gepresenteerde methode voor het semi-automatisch creëren van glossariums is het mogelijk om 63% van de definities van de vier genoemde typen automatisch te detecteren in een leertekst. Hoewel er ook een aantal niet-definities wordt geëxtraheerd, is het voor de gebruiker een eenvoudige taak om de correcte definities te selecteren en op deze manier semi-automatisch een glossarium te creëren.

Curriculum Vitae

Eline Westerhout was born in Lienden on the 21st of September, 1983. After obtaining her Atheneum degree at *Het Wartburgcollege*, she started a bachelor 'Communication studies' at the University of Tilburg with a specialization in the direction of 'Language and Artificial Intelligence'. In 2004 she obtained her bachelor's degree (cum laude). She then participated in the master program 'Language and Speech Technology' at Utrecht University and obtained her master's degree in December 2005 (cum laude) with a thesis entitled *A corpus of Dutch aphasic speech: sketching the design and performing a pilot study*. In the same month, Eline started working on the European project 'Language Technology for eLearning' (LT4eL), where she supported the project coordinator in management and coordination tasks and was also actively involved in work packages on corpus construction and annotation, keyword extraction, glossary creation, validation, and ontology development. After the project ended, she got a PhD appointment at the Utrecht Institute of Linguistics in 2008 to write a dissertation on definition extraction for the semi-automatic creation of glossaries. The results of her research are reflected in this book. In September 2009, she joined the European project 'Language Technologies for Lifelong Learning' (LTfLL).