Detecting Multiword Expressions and Named Entities in Natural Language Texts

István Nagy T.

University of Szeged

October 2014

A THESIS SUBMITTED FOR THE DEGREE OF DOCTOR OF PHILOSOPHY
OF THE UNIVERSITY OF SZEGED

Supervisors:
János Csirik, DSc
Richárd Farkas, PhD

University of Szeged
Faculty of Science and Informatics
Doctoral School of Computer Science

Contents

List of Tables

List of Figures

List of Abbreviations

CRF	conditional random fields
GS	gold standard
IE	information extraction
IR	information retrieval
LVC	light verb construction
MT	machine translation
MWE	multiword expression
NC	nominal compound
NE	named entity
NER	named entity recognition
NLP	natural language processing
NOM	nominalized light verb construction
PART	light verb construction in the form of a participle
POS	part of speech
PREP	preposition
SPLIT	split light verb constructions
SVM	Support Vector Machine
VPC	verb-particle construction

Preface

Multiword expressions (MWEs) are lexical items that can be decomposed into single words and display lexical, syntactic, semantic, pragmatic and/or statistical idiosyncrasy (Sag et al., 2002; Kim, 2008; Calzolari et al., 2002). The proper treatment of multiword expressions such as *rock 'n' roll* and *make a decision* is essential for many natural language processing (NLP) applications like information extraction and retrieval, terminology extraction and machine translation, and it is important to identify multiword expressions in context. For example, in machine translation we must know that MWEs form one semantic unit, hence their parts should not be translated separately. For this, multiword expressions should be identified first in the text to be translated.

The chief aim of this thesis is to develop machine learning-based approaches for the automatic detection of different types of multiword expressions in English and Hungarian natural language texts. In our investigations, we pay attention to the characteristics of different types of multiword expressions such as nominal compounds, multiword named entities and light verb constructions, and we apply novel methods to identify MWEs in raw texts.

In the thesis it will be demonstrated that nominal compounds and multiword named entities may require a similar approach for their automatic detection as they behave in the same way from a linguistic point of view. Furthermore, it will be shown that the automatic detection of light verb constructions can be carried out using two effective machine learning-based approaches.

István Nagy T.
Szeged, October 2014

Acknowledgements

First of all, I would like to thank my supervisors, János Csirik and Richárd Farkas for their guidance and for their useful comments and advice. They played a crucial role in turning my interest to computational linguistics, for which I am most grateful.

I would also like to thank my colleagues and friends who helped me to realize the results presented here and to enjoy my period of PhD studies at the University of Szeged. In alphabetical order they are Gábor Berend, György Móra and János Zsibrita.

I am indebted to my computer linguistic colleagues – especially to Veronika Vincze –, to whom I could always turn for advice in linguistic issues.

I would also like to thank David P. Curley for scrutinizing and correcting this thesis from a linguistic point of view.

I would like to thank my girlfriend Mariann for her endless love, support and inspiration. Last, but not least, I wish to thank my parents and my sister for their constant love and support. I would like to dedicate this thesis to them as a way of expressing my gratitude and appreciation.

This study was supported by the European Union and the State of Hungary, co-financed by the European Social Fund in the framework of TÁMOP 4.2.4. A/2-11-1-2012-0001 "National Excellence Program". I am grateful for this support, which definitely acted as an accelerator for the submission of this thesis.

Chapter 1

Introduction

1.1 Motivation

Due to the widespread usage of the sophisticated communication and mobile information devices, the amount of the publicly available information has been increasing at an extraordinary rate. A substantial part of this information is available in textual form, which is written in natural language. The manual processing of this large amount of data requires enormous human effort and financial investments, which can be supported by automatic methods. Natural language processing (NLP) is the study of mathematical and computational modelling of various aspects of natural language and the development of a wide range of computational linguistics systems.

In natural languages there are many ways to express complex human thoughts and ideas. This can be achieved by exploiting compositionality, i.e. concatenating simplex elements of language and thus yielding a more complex meaning that can be computed from the meaning of the original parts and the way they are combined. However, non-compositional phrases can also be found in languages, which are complex phrases that can be decomposed into single meaningful units, but the meaning of the whole phrase cannot (or can only partially) be computed from the meaning of its parts. Such phrases are often called multiword expressions (MWEs) and they display lexical, syntactic, semantic, pragmatic and/or statistical idiosyncrasy (Sag et al., 2002; Kim, 2008; Calzolari et al., 2002). Moreover, MWEs cannot be formulated by directly aggregating the semantics of their constituents. In other words, they are lexical items that contain space. Therefore, for the natural language processing applications that require the semantic processing of texts, the detection of multiword expressions is an indispensable part.

In this thesis, we will focus on the automatic detection of English and Hungarian multiword expressions. As MWEs occur quite frequently in both languages and their proper treatment of MWEs is essential for several natural language processing applications like information extraction and retrieval, terminology extraction and machine translation, it is important to identify multiword expressions in context. For example, in machine translation we must know that MWEs form one semantic unit, hence their parts should not be translated separately. For this, multiword expressions should be identified first in the text to be

translated.

Here, we will present different types of multiwords expressions which we will concentrate on in this thesis.

Nominal compounds (NCs) are subtypes of multiword expressions. NCs are lexical units that consist of two or more elements that exist on their own, the unit functions as a noun and it usually has some extra meaning component compared with the meanings of the original parts, such as the following English and Hungarian examples:

(a) *black sheep – fekete bárány*

(b) *stock car – marhavagon*

Light verb constructions (LVCs) are another subtype of MWEs. LVCs are verb and noun combinations in which the verb has lost its meaning to some extent and the noun is used in one of its original senses. Examples taken from English and Hungarian are shown in the following:

(a) English:
to take measure
to play a role

(b) Hungarian:
őrizetbe vesz "to take into custody"
döntést hoz "to take a decision"

Named entities (NEs) are another class of linguistic elements that require specific treatment in many NLP applications ranging from information retrieval to machine translation. A named entity is a phrase in the text which uniquely refers to an entity of the world, like the name of an organization or location. Named entities often consist of more than one word; that is, they can be viewed as a specific type of multiword expressions / nominal compounds (Jackendoff, 1997; Sag et al., 2002). Similar to multiword expressions, the meaning of multiword named entities cannot be traced back to their parts. For instance, *Ford Focus* refers to a car and has nothing to do with the original meaning of *ford* or *focus*; thus it is justifiable to treat the whole expression as one unit. NEs function as nouns just like NCs. Moreover, the linguistic relatedness is demonstrated by the fact that an NC may include an NE (***FBI*** *special agent*) and may be part of an NE (*Tallulah* ***High School***), and an NE may contain another NE (as *Oxford* and *Oxford University* in *Oxford University Press*). On the other hand, sometimes it is not unequivocal to decide whether a multiword unit is a nominal compound or a named entity (e.g. *Attorney General*). Although both nominal compounds and multiword named entities consist of more than one word, they form one semantic unit and thus, they should be treated as one unit in NLP systems. Moreover, as they behave similarly we will argue that the same methods should be applied for their automatic detection.

The main aim of this thesis is to automatically identify different types of multiword expressions in English and Hungarian raw texts. As verbal MWEs and multiword named entities are quite frequent in both languages, we seek to identify them together with English nominal compounds, and we will implement several machine learning-based approaches for these tasks.

1.2 Dissertation roadmap

Here, we will summarise our findings for each chapter of the thesis and present the connection between the publications of the author referred to in the thesis and the results described in different chapters in a table.

This thesis is organized into nine main chapters. The first, introductory chapter briefly introduces the topics addressed in this thesis.

In the second chapter, we introduce the characteristics of different types of multiword expressions. We shall also present a classification of MWEs based on their syntactic behaviour. We will introduce the characteristics of nominal MWEs such as nominal compounds and multiword named entities and also analyze the characteristics of verbal MWEs such as light verb constructions from a linguistic point of view and describe our classification of LVC phenomena.

In the third chapter, we present ten corpora that were utilized when we carried out our experiments.

In the fourth chapter, we describe the machine learning methods applied and approaches used. We present the main contributions of this thesis in the next four chapters.

In the fifth chapter, we focus on the automatic detection of English nominal compounds in running texts. We present our dictionary lookup method and machine learning-based approach to detect nominal compounds. We will also investigate how previously identified nominal compounds affect named entity recognition and vice versa, how nominal compound detection is supported by identified named entities.

In the sixth chapter, we attempt to recognize different types of named entities taken from webpages. We present three different Web Content Mining problems, namely Person Disambiguation, Researcher Affiliation Extraction and Company Contact Information Extraction, which rely heavily on Named Entity Recognition. As the majority of named entities are multiword expressions, our approach will be based on techniques that can be successfully applied in nominal compound detection as well.

In the seventh chapter, we focus on the automatic detection of English and Hungarian verbal light verb constructions in running texts. Here, we will present our conditional random fields-based tool for identifying verbal light verb constructions. Furthermore, we will also investigate the effect of simple domain adaptation techniques to reduce the gap between the different domains.

In the eighth chapter, we describe our syntax-based method used to identify each LVC in English and Hungarian running texts. As we saw in Chapter 7, the CRF-based tool is able to recognize verbal LVCs, but this approach could not handle other types of LVCs. However, our goal here is to identify each LVC occurrence in running texts and mark it with

			Chapter			
			5	6	7	8
RANLP	2011	(Nagy T. et al., 2011a)	•			
RANLP	2011	(Vincze et al., 2011b)	•			
MWEWS	2011	(Vincze et al., 2011a)	•			
TSD	2013	(Nagy and Vincze, 2013)	•			
ACTA	2012	(Nagy T., 2012)		•		
NLPIR4DL	2009	(Nagy et al., 2009)		•		
OTDK	2009	(Nagy T., 2009)		•		
ACM	2013	(Vincze et al., 2013b)			•	
ACL	2013	(Vincze et al., 2013a)				•
IJCNLP	2013	(Nagy T. et al., 2013)				•

Table 1.1: The relation between the thesis topics and the corresponding publications.

our syntax-based method.

In the final chapter we provide a brief summary of the whole thesis both in English and Hungarian. Table 1.1 summarises the relationship among the thesis chapters and the more important referred publications.

Chapter 2

The Characteristics of Multiword Expressions

Multiword expressions can be divided into several groups (Vincze, 2011; Kim, 2008), based on the parts of speech of their components or based on their syntactic and semantic behaviour. In this chapter, we present a classification of MWEs based on their syntactic behaviour. First, the characteristics of nominal MWEs such as nominal compounds and multiword named entities are examined. Then, verbal MWEs such as light verb constructions, verb-particle constructions and idioms are investigated followed by other types of MWEs. In the thesis, we focus on the automatic detection of English and Hungarian MWEs, so we illustrate the above types of MWEs in these two languages. As LVCs and multiword named entities are quite frequent in both languages, we aim at identifying them together with English nominal compounds and they will be characterised in more detail.

2.1 Nominal Multiword Expressions

In this section we will introduce the different types of nominal MWEs.

2.1.1 Nominal Compounds

A compound is a lexical unit that consists of two or more elements that exist on their own (Sag et al., 2002; Kim, 2008). They can function as adjectives (*Roman Catholic*), prepositions (*in front of*), conjunctions (*in order to*) and nouns (*lunch time*), however, here we just concentrate on nominal compounds. Orthographically, compounds may include spaces (*swimming pool*, *fekete doboz* "black box") or hyphens (*self-esteem*, *természetesnyelv-feldolgozás* "natural language processing") or none of them (*blackboard*, *autópálya* "highway") (Vincze et al., 2011b). The characteristic on nominal compounds will described in Section 2.4.

2.1.2 Multiword Named Entities

Many times named entities consist of more than one words, numbers or even characters. For instance, the official person names (*Mariann Majer*) consist of first names and last names, while the names of organisations (*University of Szeged*, *Flow2000 Bt.*) indicate the types of the organisations like *institute* or *ltd.* and they usually have a unique name. Also, addresses usually contain the name of the city, zip code of the city and the name of the street, like *6720 Szeged, Dugonics square 13*.

It is often the case that their meaning cannot be traced back to their parts. For instance, *Ford Focus* refers to a car and has nothing to do with the original meaning of *ford* or *focus*. Therefore, it is necessary to treat multiword named entities as one unit.

2.2 Verbal Multiword Expressions

Here, we will present the different types of verbal MWEs.

2.2.1 Verb–particle Constructions and Verbs with Prefixes

Verb–particle constructions (also called phrasal verbs or phrasal–prepositional verbs) are made up of a verb part and a particle/preposition part (see i.e. Kim (2008),Vincze et al. (2011b)). They can be adjacent (as in ***put off*** *the meeting*) or separated by an intervening object (***turn*** *the light* ***off***). Their meaning can be compositional, i.e. it can be computed from the meaning of the preposition and the verb (*lie down*) or non-compositional (*do in* "kill").

In Hungarian, verbs can have verbal prefixes, which can be separated from the verb for syntactic reasons (compare *eldobja* "he throws it away" and *nem dobja el* "he does not throw it away"). In this case, the meaning of the verb with the prefix is compositional, but other examples such as *kinyír* (out.cut) "kill" are idiomatic.

2.2.2 Idioms and Proverbs

An idiom is a MWE whose meaning cannot (or can only partially) be determined on the basis of its components (Sag et al., 2002; Nunberg et al., 1994). Although most idioms behave normally as far as the morphology and syntax are concerned, i.e. they can undergo some morphological change (i.e. verbs are inflected in a normal way as in *He spills/spilt the beans*), their semantic aspect is totally unpredictable.

Proverbs express some important facts that are thought to be true by most people. Proverbs usually take the same form and show no morphological change (i.e. they are fixed expressions). Some examples are: *An apple a day keeps the doctor away* or *You can catch more flies with honey then you can with vinegar* or in Hungarian: *Ahány ház, annyi szokás* "each house has its traditions" or *Addig jár a korsó a kútra, míg el nem törik* "destiny reaches everyone".

2.2.3 Light Verb Constructions

Light verb constructions consist of a nominal and a verbal component, where the noun is usually taken in one of its literal senses, but the verb usually loses its original sense to some extent (Stevenson et al., 2004; Vincze, 2011). Some examples are offered here: *take a decision – döntést hoz*, *make a contract – szerződést köt*.

Light verb constructions are syntactically flexible, that is, they can manifest in various forms: the verb can be inflected, the noun can occur in its plural form, or the noun can be modified. The nominal and the verbal component may not be adjacent in the sentence as in:

(2.1) Ő kötötte azzal a céggel azt az előnyös szerződést.
"It was him who made that beneficial contract with that company."

The characteristic of light verb constructions will be presented in Section 2.5.

2.3 Other Types of Multiword Expressions

There are other types of MWEs that do not fit into the above categories (some of them are listed in Jackendoff (1997) and Vincze et al. (2011b)). Determinerless PPs are made up of a preposition and a singular noun (without a determiner) (Kim, 2008) – in this way, they are syntactically marked – and they usually function as an adverbial modifier (*in case*, *for good*, *on foot* etc.).

Another group of MWEs is formed of foreign phrases such as *status quo*, *c'est la vie* and *ad hoc*. Although they are composed of perfectly meaningful parts in the original language, these words do no exist on their own in English, hence it is impossible to derive their meaning from their parts and the expression must be stored as a whole.

More complex and longer MWEs are quotations ("May the Force be with you"), lyrics of songs, clichés and commonplaces (*That's life*) are also similar to them in that they are longer MWEs and are not changeable (see Jackendoff (1997) for details).

2.4 The Characteristics of Nominal Compounds

Nominal compounds (NCs) form a subtype of multiword expressions: they form one unit the parts of which are meaningful units on their own, the unit functions as a noun and it usually has some extra meaning component compared with the meanings of the original parts (Sag et al., 2002; Kim, 2008). The semantic relation between the parts of the nominal compound may vary: it may express a "made of" relation (*apple juice*), a "location" relation (*neck pain*) or a "made for" relation (*hand cream*) just to name a few. Thus, nominal compounds encode some important meaning components that can be fruitfully applied by e.g. information extraction systems. However, such applications require that nominal compounds should be previously known to the system.

Nominal compounds occur frequently in everyday English (in the Wiki50 corpus (Vincze et al., 2011b), 67.3% of the sentences on average contain a nominal compound). Furthermore, they are productive: new nominal compounds are entering the language all the time,

hence they cannot be exhaustively listed and appropriate methods should be implemented for their identification.

It is also important to emphasize that a nominal compound candidate does not always function as a nominal compound. Take, for instance, *tall boy*: when it refers to a can of beer, it is an MWE, but when it refers to a young male of somewhat unusual height, it is simply a productive combination of an adjective and a noun and does not constitute an MWE. Thus, nominal compounds should be identified in context, i.e. in running texts, and we will follow this approach in our investigations.

2.5 The Characteristics of Light Verb Constructions

Light verb constructions are verb and noun combinations where the semantic head of the construction is the noun, i.e. the verb has lost its meaning to some extent and the noun is used in one of its original senses, but the verb functions as the syntactic head (the whole construction fulfills the role of a verb in the clause) (Vincze, 2011).

Light verb constructions exhibit lexical and semantic idiosyncracy (to some extent). As for the former, the verbal component of the construction cannot be substituted by another verb with a similar meaning: instead of *make a decision* we cannot say **do a decision*. Still, the change of the noun for a word with a similar meaning does not yield the agrammaticality of the construction: *make a contract* and *make a treaty* are both acceptable constructions. Next, it should also be mentioned that there seem to be systematic cases where two light verb constructions share all of their meaning components but their verbal components differ. Take, for instance:

(2.2) make/take a decision

With regard to semantic idiosyncracy, the meaning of light verb constructions can, at least partially, be computed from the meanings of their parts and the way they are connected. Although it is the noun that conveys most of the meaning of the construction, the verb itself cannot be viewed as semantically bleached (see i.e. Apresjan (2004), Alonso Ramos (2004), Sanromán Vilas (2009)) since it also adds important aspects to the meaning of the construction. For instance, (2.3) and (2.4) do not mean the same though they describe the same situation of helping:

(2.3) give help

(2.4) receive help

Light verb constructions are syntactically flexible, that is, they can manifest themselves in a variety of forms: the verb may be inflected, the noun may occur in its plural form and the noun may be modified. The nominal and the verbal component may not be adjacent in the sentence as in:

(2.5) The **decision** he **took** last time proved to be fatal.

The above points have some consequences for the NLP treatment of light verb constructions. Syntactic flexibility makes the automatic identification of light verb constructions difficult, especially in the case of agglutinative languages such as Hungarian. Lexical and semantic idiosyncracy can also affect the machine translation of the constructions: the nominal component, being the semantic centre of the construction, seems to be constant across languages, hence it can be translated literally whereas the verb can be determined only lexically, i.e. they must be sorted in dictionaries, cf. *take a decision* and *döntést hoz*, which literally means "bring a decision".

2.5.1 Light Verb Constructions in Hungarian

In order to understand the special features of identifying Hungarian light verb constructions, a brief description of the Hungarian language is required. Hungarian is an agglutinative language, which means that a word can have hundreds of word forms due to inflectional or derivational morphology (É. Kiss, 2002). Hungarian word order is related to information structure, e.g. new (or emphatic) information (focus) always precedes the verb and old information (topic) precedes the focus position. Thus, the position relative to the verb has no predictive force as regards the syntactic function of the given argument. In English, the noun phrase before the verb is most typically the subject whereas in Hungarian, it is the focus of the sentence, which itself can be the subject, object or any other argument.

The grammatical function of words is determined by case suffixes. Hungarian nouns can have about 20 cases, which mark the relationship between the verb and its arguments (subject, object, dative, etc.) and adjuncts (mostly adverbial modifiers). Although there are postpositions in Hungarian, case suffixes can also express relations that are expressed by prepositions in English. As for verbs, they are inflected for person and number and the definiteness of the object.

The canonical form of a Hungarian light verb construction is a bare noun + third person singular verb. Due to the above features, they may occur in non-canonical versions as well: the verb may precede the noun, or they may be not adjacent. Moreover, the verb may occur in different surface forms inflected for tense, mood, person and number. These issues will be considered when implementing our system for identifying Hungarian light verb constructions.

2.5.2 Types of Light Verb Constructions

The presentation of the types of light verb constructions is based on Vincze (2011) and Vincze et al. (2013b). The papers present a test battery which is able to differentiate among different types of verb + noun combinations: productive constructions, light verb constructions and idioms. Two tests, namely the tests of variativity and omitting the verb play the most significant role in distinguishing LVCs from productive constructions and idioms. Variativity reflects the fact that LVCs can be often substituted by a verb derived from the same root as the nominal component within the construction: productive constructions and idioms can be rarely substituted by a single verb. Even if so, there is no morphological relation between the noun and the verbal counterpart. Omitting the verb exploits the fact that

Test	Productive	LVC		Idiom
		productive-like	idiom-like	
WH-word	YES	YES	NO	NO
Article	YES	YES	NO	NO
Plural	YES	YES	NO	NO
Negation	YES	YES	NO	NO
Possessor	YES	YES	NO	NO
Attributive	YES	YES	NO	NO
Coordination	YES	NO	NO	NO
Nominalization (V)	NO	NO	YES	NO
Nominalization (LVC)	YES	YES	NO	NO
Participle – 1	YES	YES	YES	YES
Participle – 2	YES	YES	NO	NO
Variativity	NO	YES	YES	NO
Changing the verb	YES	YES	NO	NO
Omitting the verb	NO	YES	YES	NO

Table 2.1: Tests for differentiating productive constructions, light verb constructions and idioms.

it is the nominal component that mostly bears the semantic content of the LVC, hence the event denoted by the construction can be determined even without the verb in most cases. Both the noun and the verb play a key role in computing the meaning of productive constructions, while the original senses of the noun and the verb are not relevant at all as regards the meaning of an idiomatic verb + noun combination. Thus, the noun itself is not sufficient to compute the meaning of either productive or idiomatic constructions.

The other tests help us to distinguish between two types of light verb constructions. Productive-like LVCs behave rather like productive constructions, whereas idiom-like constructions are more similar to idioms. Still, there is no sharp or distinct boundary in between the groups since belonging to a (sub)group is not determined by a dichotomy of the either-or type: the place of the construction on a scale is rather a question of degree and scalability, which is true for English and Hungarian as well (Vincze, 2011). Table 2.1 states the applicability of the tests for each type and these tests were used in annotating the corpora presented in Sections 3.5, 3.6, 3.7, 3.10, 3.11.

Krenn (2008) provides some diagnostic tests for distinguishing between German idioms and light verb constructions. As for English, Kearns (2002) distinguishes between two subtypes of what is traditionally called light verb constructions. True light verb constructions such as *to give a wipe* or *to have a laugh* and vague action verbs such as *to make an agreement* or *to do the ironing* differ in some syntactic and semantic features and can be separated by various tests (e.g. passivization, WH-movement, pronominalization, etc.), as shown in Table 2.2. True light verb constructions roughly correspond to idiom-like LVCs in Vincze's (2011) classification, whereas vague action verbs are similar to productive-like constructions. Examples for the above types of light verb constructions can be seen in Figure 2.1.

From a morphological perspective, light verb constructions can also be divided into

Test	Vague action verb	True light verb
Passivization	YES	NO
WH-movement	YES	NO
Pronominalization	YES	NO
Indefinite NP	NO	YES
NP stem is identical to a verb	NO	YES
Differences compared to verbal counterpart	NO	YES
Examples	make an inspection	give a groan

Table 2.2: True light verbs and vague action verbs in English.

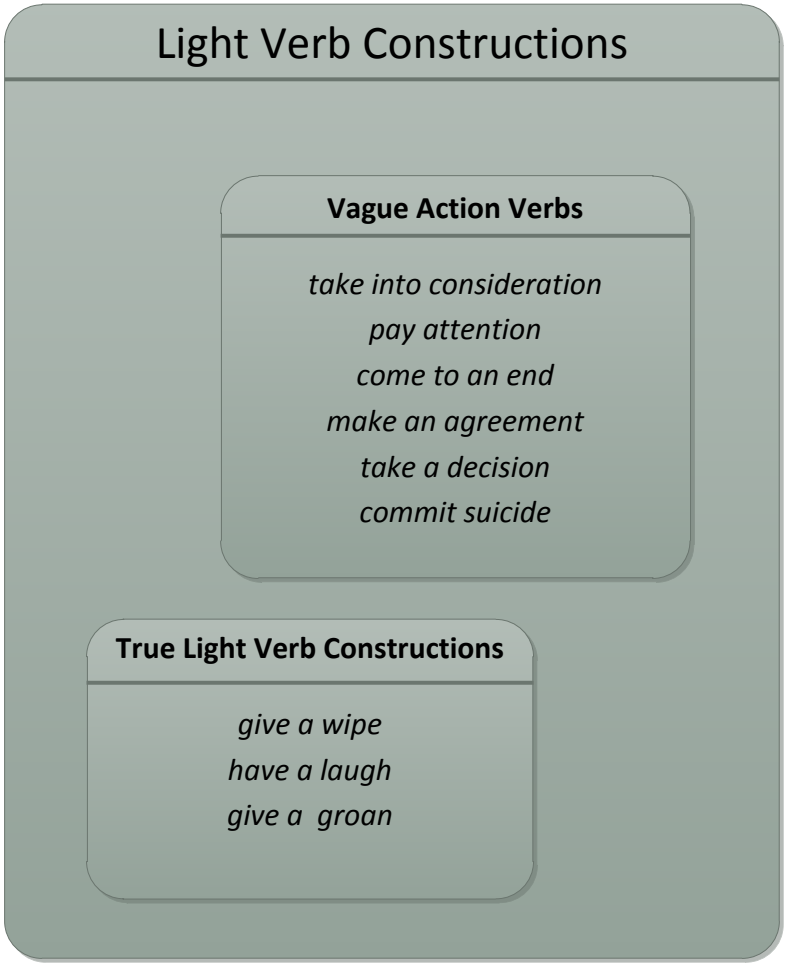

Figure 2.1: Types of light verb constructions based on syntactic and semantic criteria.

groups. First, the most common type is when the nominal component is the object of the verb, i.e. it bears an accusative case in Hungarian. Second, the nominal component can bear other (oblique) cases as well in Hungarian. (This option is not viable in English, due to the lack of oblique morphological cases.) Third, a prepositional or postpositional phrase can also occur in the construction. Figure 2.2 presents this classification with illustrative examples.

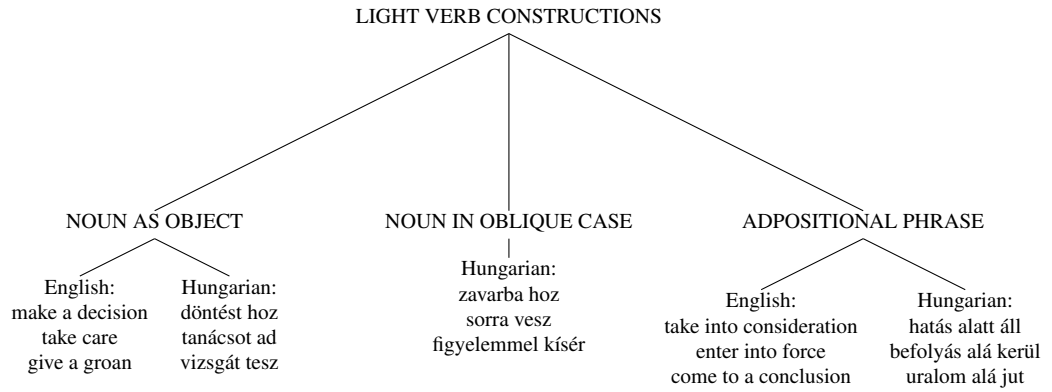

Figure 2.2: Types of light verb constructions seen from a morphological point of view.

Light verb constructions may occur in several forms due to their syntactic flexibility. Besides the prototypical verb + noun combination in English (VERB, e.g. *make contracts*) and the noun + verb combination in Hungarian (e.g. *szerződést köt*), they can have a participial form (PART, e.g. *contracts made*) and they may also undergo nominalization, yielding a nominal compound (NOM, e.g. *contract maker*). In split light verb constructions (SPLIT, e.g. *a contract which has been recently made*), the noun and the verb may be situated far from each other in the sentence, so the construction is non-contiguous, therefore their identification may require going beyond clause boundaries.

2.6 Summary

In this chapter we described the multiword expressions which we will focus on in this thesis. We will automatically detect English nominal compounds in Chapter 5 and present our methods to extract English and Hungarian multiword Named Entities from webpages in Chapter 6. Finally, we will identify English and Hungarian LVCs in Chapters 7 and 8.

Chapter 3

Applied corpora

3.1 Introduction

Manually annotated corpora are required to apply supervised machine learning-based methods for the automatic detection of multiword expressions and named entities from running text. First we present corpora annotated for named entities and multiword expressions then we will pay special attention to manually annotated corpora that will be used when we conduct our experiments on detecting multiword expressions and named entities in Chapters 5, 6, 7 and 8.

As for named entities, several corpora have been constructed, for instance, within the framework of the ACE project (Doddington et al., 2004) and for international challenges such as the CoNLL-2002/2003 datasets (Tjong Kim Sang, 2002; Tjong Kim Sang and De Meulder, 2003) or the MUC datasets (Grishman and Sundheim, 1995; Chinchor, 1998) – just to name a few. CoNLL-2003 norms were followed when constructing named entity corpora for Hungarian business news (Szarvas et al., 2006a) and criminal news[1]. Manual annotation is costly and time consuming, so there have been some attempts to automatically generate silver standard NER corpora both in English and Hungarian (Nemeskey and Simon, 2012).

Identifying multiword expressions is not unequivocal since constructions with a similar syntactic structure (e.g. verb + noun combinations) may belong to different subclasses on the productivity scale (i.e. productive combinations, light verb constructions and idioms, see Fazly and Stevenson (2007)). This is why well-designed and tagged corpora of multiword expressions are invaluable resources for training and testing algorithms that are able to identify multiword expressions.

Several corpora and databases of MWEs were constructed for a number of languages. For instance, Nicholson and Baldwin (2008) describe a corpus and a database of English compound nouns (BNC dataset in Section 3.9). Electronic databases for French multiword nouns and adverbs (Laporte et al., 2008; Laporte and Voyatzi, 2008) and German adjective+noun collocations (Evert, 2008) were also created and lexicons of (idiomatic) multiword units were developed for Dutch (Grégoire, 2007; Grégoire, 2010) and German and English (Anastasiou and Carl, 2008).

[1] goo.gl/Cq0eXV

An Estonian database and a corpus of multiword verbs were constructed (Kaalep and Muischnek, 2006; Kaalep and Muischnek, 2008; Muischnek and Kaalep, 2010)) and Krenn (2008) developed a database of German PP-verb combinations. The Prague Dependency Treebank was also annotated for multiword expressions (Bejcek and Stranák, 2010) and light verb constructions (Cinková and Kolářová, 2005). For Portuguese, Hendrickx et al. (2010) created an annotated corpus of complex predicates (i.e. multiword verbs), and Sanches Duran et al. (2011) presented a dictionary of Brazilian Portuguese complex predicates. NomBank (Meyers et al., 2004) contains the argument structure of common nouns, paying attention to those occurring in light verb constructions as well. Literal and idiomatic uses of English verb + noun combinations are annotated in the VNC-Tokens dataset (Cook et al., 2008).

3.2 The English Name Disambiguation Test Corpus

The Web People Search task has been defined in WePS campaigns as the problem of organizing web search results for a given person name. The WePS campaign (Artiles et al., 2010) introduced a task which sought to mine attributes for persons, i.e. rather than recognizing attributes in webpages, the task was to assign them to people (the clusters of pages belonging to each given person). As these person related attributes are named entities, it was treated as a Named Entity Recognition problem.

The third Web People Search (WePS3) (Artiles et al., 2010) datasets were used for testing our English name disambiguation approach described in Section 6.5. It contains 300 names, in contrast to WePS2 (Artiles et al., 2009) where the test database consisted of only 30 names. The person names were obtained from a US Census, Wikipedia, Computer Science PC lists and names which contained at least one person who was a lawyer, corporate executive or estate agent. For each name the top 200 web search results from the Yahoo! API were downloaded and archived with their corresponding search metadata, like search snippet, title, URL and position in the results ranking.

During the annotation process, only websites that were related to one of two predefined persons were labeled by the annotators. In this way, the annotation effort was radically reduced. Consequently, large amounts of human resources and time were saved. Clearly, the gold standard used was not perfect.

The attribute extraction subtasks were not manually annotated in the gold standard test database of WePS3. The systems outputs were manually evaluated by annotators. They got a website with the ten most common attribute–value pairs according to the participant systems, and he or she had to decide which attribute belonged to which of the following categories:

- Correct: the attribute appears in the website and it is related to the actual person.
- Incorrect for any reason other than being too long or too short. For instance, the type of attribute is incorrect (e.g. a number is incorrectly identified as a telephone number); the attribute is not related to the actual person (e.g. the attribute describes some other person described on the page); or the attribute simply did not appear in the text.
- The attribute is correct, but it is too long or too short. So the attribute has one of the following problems:

  - It is too short. The attribute is incomplete (e.g. *director* when it should say *director of marketing*).
  - It is too long. The attribute contains a correct value, but includes irrelevant information (e.g. *CEO in 1982* when it should say just *CEO*).

- Cannot decide because the webpage is unreadable for some reason.
- The webpage is readable, but the specified person is not on this page.

We used the datasets made available by the shared task organisers at `goo.gl/vUWuU9` to train and evaluate our automatic English name disambiguation system.

3.3 Hungarian Company Contact Information Web Corpus

The Hungarian Company Contact Information Corpus with manually annotated webpages was used to evaluate our information extraction system described in Nagy T. (2009). In the corpus, the names and the addresses of Hungarian companies were manually annotated. It consists of 100 randomly selected Hungarian companies taken from the database of cylex.hu and 454 corresponding webpages where the names and the addresses of companies are available. During the annotation process a Firefox extension (Farkas et al., 2008) was used to manually label the different annotated categories. The average agreement rate among the annotations was an F-score of 82.63. Table 3.1 summarizes the number of occurrences for each annotated category.

Attribute	**Occurrence**
house number	515
ZIP code	436
street	526
city	536
company	936

Table 3.1: Number of occurrences of categories in the Hungarian Company Contact Information Corpus.

The corpus is available under the Creative Commons licence at `goo.gl/NBMxBT`.

3.4 Researcher Affiliation Corpus

The Researcher Affiliation Corpus (Farkas et al., 2008) is a manually constructed webpage corpus containing HTML documents annotated for publicly available information about researchers. It contains 455 sites, 5282 pages for 89 researchers (who form the Programme

Committee of the SASO07 conference[2]). The corpus is extensively annotated, has a three-level deep annotation hierarchy with 44 classes (labels).

However, only one particular information class, namely affiliation was targeted in our investigations. The `affiliation` is defined as the current and previous physical workplaces and higher educational institutes of the researcher. Here institutes related to review activities, awards, or memberships are not regarded as affiliations. The `position` is regarded as the tuple of <`affiliation`, `position types`, `years`>, as for example in <*National Department of Computer Science and Operational Research at the University of Montreal, adjunct Professor, 1995, 2002*>[3]. Among the four slots just the affiliation slot is mandatory (it is the head) as the others are usually missing in real homepages.

The corpus is publicly available under the Creative Commons licence at `goo.gl/c3pl9F`.

3.5 The Wiki50 Corpus

The Wiki50 corpus (Vincze et al., 2011b) consists of 50 randomly selected articles taken from the English Wikipedia. The only selectional criterion applied was that each article should consist of at least 1000 words and they should not contain lists, tables or other structured texts (i.e. only running texts were included).

In the corpus, several types of multiword expressions and four classes of named entities (NEs) were manually annotated. In the case of nominal compounds, only the compounds with spaces were annotated since hyphenated compounds (e.g. *self-esteem*) can be easily recognized. The annotation of light verb constructions was based on the test battery described in Section 2.5.2, but no subtypes of LVCs are distinguished (i.e. vague action verbs and true light verb constructions are annotated in the same way). In the case of named entities, tag-for-meaning annotation was applied as occurrences of e.g. country names for instance might refer to an organization and a location as well, depending on the context. The corpus contains 114,570 tokens in 4,350 sentences. Table 3.2 summarizes the number of occurrences and the number of unique phrases (i.e. no multiple occurrences are counted here) for each annotated category. The corpus contains 368 occurrences of 287 light verb constructions and 2929 occurrences of 2405 nominal compounds.

Fifteen articles were annotated by all the annotators of the corpus. As for nominal compounds and light verb constructions, the agreement rates between the two annotators were 71.1 and 70.7 (F-score) and 51.98 and 54.26 (Jaccard), respectively. The corpus is available under the Creative Commons licence at `goo.gl/TvSK05`.

3.6 The Szeged TreebankFX

The Szeged Treebank (Csendes et al., 2005) is the biggest manually annotated Hungarian corpus available to date. It is a morphosyntactically tagged and syntactically annotated database, which is available in both constituency-based (Csendes et al., 2005) and dependency-

[2] goo.gl/cv4uW1

[3] The example is extracted from goo.gl/Gcf9mZ.

Category	Occurrence	Unique phrases
Nominal compound	2929	2405
Adjectival compound	78	60
Verb–particle combination	446	342
Light verb construction	368	338
Idiom	19	18
Other MWE	21	17
MWEs total	3861	3180
Person	4093	1533
Organization	1498	893
Location	1558	705
Miscellaneous NE	1827	952
NEs total	8976	4083

Table 3.2: Identified occurrences of categories in the Wiki50 corpus

based (Vincze et al., 2010) versions. In the corpus, each word is assigned all its possible morphosyntactic tags and lemmas and the appropriate one is selected according to the context. The genres included in the Szeged Treebank are the following:

- **Business news:** Short (1 or 2 sentences long) pieces of news taken from the archive of the Hungarian News Agency.
- **Newspaper articles:** Excerpts from three daily papers (*Népszabadság*, *Népszava* and *Magyar Hírlap*) and one weekly paper (*HVG*).
- **Legal texts:** Excerpts from laws on economic enterprises and authors' rights.
- **Fiction:** Three novels: Jenő Rejtő: *Piszkos Fred, a kapitány* (Dirty Fred, the Captain), Antal Szerb: *Utas és holdvilág* (Journey by Moonlight) and George Orwell: *1984*.
- **Computer-related texts:** Excerpts from Balázs Kis: *Windows 2000 manual book* and some issues of the *ComputerWorld: Számítástechnika* magazine.
- **Composition:** Short essays written by 14-16-year-old students.

	Sentences	Tokens
Business news	9,574	227,239
Newspapers	10,210	223,286
Legal texts	9,278	258,722
Fiction	18,558	237,741
Computer texts	9,627	214,803
Composition	24,720	343,010
Total	81,967	1,504,801

Table 3.3: Statistical data on the Szeged Treebank corpus.

	VERB	PART	NOM	SPLIT	total
Business news	565	697	90	40	1392
	40.6%	50.1%	6.5%	2.9%	20.7%
Newspapers	458	197	55	67	777
	58.9%	25.4%	7.1%	8.6%	11.5%
Legal texts	641	679	710	241	2,271
	28.2%	29.9%	31.3%	10.6%	33.7%
Fiction	567	61	5	93	726
	78.1%	8.4%	0.7%	12.8%	10.8%
Computer texts	429	126	85	76	716
	59.9%	17.6%	11.9%	10.6%	10.6%
Composition	582	122	9	139	852
	68.3%	14.3%	1.1%	16.3%	12.7%
Total	3,242	1,882	954	656	6,734
	48.1%	27.9%	14.2%	9.7%	100%

Table 3.4: Subtypes of light verb constructions in the Szeged Treebank. **VERB:** verbal occurrences. **PART:** participial light verb constructions. **NOM:** nominal light verb constructions. **SPLIT:** split light verb constructions.

Subcorpus	**VERB**		**PART**		**NOM**		**SPLIT**		**Total**		**LVC/SAU%**		**SAUs**
	EN	**HU**	**EN**	**HU**	**EN**	**HU**	**EN**	**HU**	**EN**	**HU**	**EN**	**HU**	
EU	132	158	30	76	24	32	41	29	227	295	14.95	19.43	1518
Magazines	356	387	55	120	31	42	83	53	525	602	9.87	11.32	5320
Language book	158	79	5	21	14	4	22	15	199	119	5.69	3.4	3496
Literature	270	261	15	24	6	5	119	57	410	347	12.69	10.74	3232
Miscellaneous	7	12	1	1	1	0	1	1	10	14	1.44	2.01	695
Total	923	897	106	242	76	83	266	155	1371	1377	9.61	9.66	14261

Table 3.5: Subtypes of English/Hungarian light verb constructions in SzegedParalellFX. **VERB:** verbal occurrences. **PART:** participial light verb constructions. **NOM:** nominal light verb constructions. **SPLIT:** split light verb constructions. **SAU:** Number of sentence alignment units.

All the subcorpora were annotated for LVCs on the basis of the test battery described in Section 2.5.2, but no subtypes of LVCs are distinguished (i.e. vague action verbs and true light verb constructions are annotated in the same way). The corpus contains 6,734 light verb constructions in 82,099 sentences. Statistical data on the Szeged Treebank can be seen in Tables 3.3 and 3.4 based on Vincze (2011). The corpus is publicly available for research and/or educational purposes under the Creative Commons licence at `www.inf.u-szeged.hu/rgai/mwe`.

3.7 The SzegedParalellFX Corpus

The SzegedParalell English–Hungarian parallel corpus (Tóth et al., 2008) contains texts got from the following domains:

- **Language book sentences**: This subcorpus comprises sentences taken from language books of English for Hungarian learners.
- **Texts on the European Union**: This subcorpus comprises texts collected from the `https://europa.eu.int` website. Topics include the history of the European Union and the monetary system of the EU.
- **Bilingual magazines**: This subcorpus consists of texts taken from the bilingual in-flight magazine, *Horizon Magazine* of Malév (Hungarian Airlines) and texts taken from a bilingual newspaper on real estate (*Resource Ingatlan Info*).
- **Literature**: Novels and short stories were mainly collected from the Hunglish corpus (Halácsy et al., 2005) and the Hungarian Electronic Library (`goo.gl/3NTKwA`).
- **Miscellaneous texts**: Some short texts (e.g. recipes) were also collected from the internet and included in the corpus.

However, not the whole SzegedParalell corpus was annotated for light verb constructions, with only three novels annotated. The resulting corpus is called SzegedParalellFX (Vincze, 2012) and it consists of 14,261 sentence alignment units (SAUs). The total number and the number of the subtypes of light verb constructions are presented in Table 3.5 based on Vincze (2012).

In the Hungarian part of the corpus, there are 1,371 occurrences of 672 light verb constructions in 14,261 sentence alignment units, so, a specific light verb construction occurs 2.05 times on average in the corpus. As for the English data, 706 light verb constructions occur altogether 1,371 times (1.94 times each on average).

In order to measure inter annotator agreement rate, 928 sentence alignment units were annotated by all the annotators. The average agreement rate was 78.15 on the English data and 74.23 on the Hungarian data (agreement rates are given in terms of F-score). The corpus is available under the Creative Commons licence at `goo.gl/qASBjm`.

3.8 The Tu&Roth Dataset

In the Tu&Roth dataset (Tu and Roth, 2011), English true light verb constructions were annotated. This dataset consists of 2,162 sentences randomly selected from the British National Corpus that contain verb-object pairs formed with the verbs *do*, *get*, *give*, *have*, *make* and *take*. In this case, positive and negative examples were also marked and the different examples were balanced (1,039 positive and 1,123 negative examples). Furthermore, since the corpus was created by collecting sentences that contain verb-object pairs with specific verbs, this dataset contains a lot of negative and ambiguous examples besides annotated LVCs, hence the distribution of LVCs in the Tu&Roth dataset is not comparable to those in Wiki50 or SzegedParalellFX. In this dataset, only one positive or negative example was annotated in each sentence, and they examined just the verb-object pairs formed with the six verbs as a potential LVC. However, the corpus probably contains other light verb constructions which were not annotated. For example, in the sentence *it have (sic!) been held that*

Corpus	Sentences	Tokens	VERB		PART		NOM		SPLIT		Total
			#	%	#	%	#	%	#	%	
JRC-Acquis	5,619	103,963	204	41.9	157	32.2	24	4.9	102	21	487
CoNLL-2003	8,467	107,620	235	59.2	83	20.9	16	4	63	15.9	381

Table 3.6: Statistical data on the JRC-Acquis and CoNLL-2003 corpora. **VERB:** verbal occurrences. **PART:** participial light verb constructions. **NOM:** nominal light verb constructions. **SPLIT:** split light verb constructions.

a gift to a charity of shares in a close company gave rise to a charge to capital transfer tax where the company had an interest in possession in a trust, the phrase *give rise* was listed as a negative example in the Tu&Roth dataset, but *have an interest*, which is another light verb construction was not marked as either positive or negative. The corpus is available at `goo.gl/K3939W`.

3.9 The BNC Dataset

The BNC dataset consists of 1000 sentences taken from the British National Corpus that contains 485 two-part nominal compounds (Nicholson and Baldwin, 2008). The dataset includes texts from various domains such as literary works, essays and newspaper articles. The corpus contains some annotation errors, like marking nominal compounds that contain a proper noun, e.g. *Belfast primary school headmaster*, as simple nominal compounds instead of proper nouns (as they should be according to the guidelines). The BNC dataset is available under the Creative Commons licence at `goo.gl/lHPhyQ`.

3.10 The JRC-Acquis Corpus

The JRC-Acquis Multilingual Parallel Corpus consists of legislative texts for a range of languages used in the European Union (Steinberger et al., 2006). 60 randomly selected documents were taken from the English version of the corpus and LVCs in them were annotated. The annotation guidelines used were the same as those used for Wiki50 and SzegedParalellFX corpora. The corpus contains 103,963 tokens in 5,619 sentences and 487 occurrences of manually annotated LVCs. Statistical data on the corpus can be seen in Table 3.6. The JRC-Acquis corpus is available under the Creative Commons licence at `goo.gl/CxedMi`.

3.11 The CoNLL-2003 Corpus

The CoNLL-2003 dataset (Sang and Meulder, 2003) was originally developed for Named Entity Recognition in the short news domain. 500 randomly selected pieces of short news were taken from the CoNLL-2003 dataset and LVCs in them were annotated. The annotation process was the same as in Wiki50 and SzegedParalellFX corpora. This corpus contains 381

Corpus	English	Hungarian	LVC	NC	NE
WePS3	•				•
Company Contact Information Corpus		•			•
Researcher Affiliation Corpus	•				•
Wiki50	•		•	•	•
SzegedParalellFX	•	•	•		
Tu&Roth	•		•		
BNC	•			•	
Szeged TreebankFX		•	•		
CoNLL-2003	•		•		
JRC-Acquis	•		•		

Table 3.7: Features of the corpora

Corpus	Sentence	Token	LVC	NC
Wiki50	4,350	114,570	368	2,929
SzegedParalellFXEng	14,262	298,948	1,371	–
SzegedParalellFXHu	14,528	250,129	1,377	–
Tu&Roth	2,162	65,060	1,039	–
BNC	1,000	21,631	–	368
Szeged TreebankFX	81,967	1,504,801	6,734	–
CoNLL-2003	8,467	107,620	381	–
JRC-Acquis	5,619	103,963	487	–

Table 3.8: Number of sentences, words, nominal compounds and light verb constructions on different corpora

occurrences of manually annotated LVCs in 8,467 sentences. Table 3.6 lists the statistical data on the corpus. The CoNLL-2003 corpus is available under the Creative Commons licence at `goo.gl/CxedMi`.

Tables 3.7 and 3.8 list the key statistical data for different corpora. Moreover, Table 3.9 summaries the relationship among the thesis chapters and corpora applied.

3.12 Corpus Roadmap

In the fifth chapter we focus the automatic detection of nominal compounds in raw text. Here, the Wiki50 corpus and BNC datasets were applied, as nominal compounds were manually annotated in these two corpora.

Named Entity Recognition methods presented in the sixth chapter are based on The English Name Disambiguation Test Corpus, Hungarian Company Contact Information Web Corpus and Researcher Affiliation Corpus.

As the seventh chapter demonstrates how portable our machine-learning-based models are among different domains in two typologically different languages, several corpora were used. In this case, when choosing the corpora we kept in mind the fact that the same domains

Corpus	5	6	7	8
WePS3		•		
Companies Contact Information Corpus		•		
Resercher Affiliation Corpus		•		
Wiki50	•			
SzegedParalellFX			•	•
Tu&Roth			•	•
BNC	•			
Szeged TreebankFX			•	•
CoNLL-2003			•	
JRC-Acquis			•	

Table 3.9: The relation between the thesis chapters and the corresponding corpora.

would be employed for both languages. Here the English part of the SzegedParalellFX, JRC-Acquis and CoNLL-2003 corpora were used in English and the subcorpora of Szeged TreebankFX in Hungarian.

As we focus on the full-coverage detection of light verb constructions in Chapter 8, the Wiki50 and SzegedParalellFX full-coverage annotated corpora were applied, where each type and individual occurrence of a light verb construction was marked in running texts.

Chapter 4

Machine Learning Techniques

4.1 Introduction

In this chapter, we will provide a brief overview of common notations and definitions for machine learning. We will present some basic methods in machine learning and then we will describe the most important issues in evaluation methodology.

4.2 Basic Concepts of Machine Learning

The machine learning task generally consists of N number of different objects (they are also called *instances* or *entities*) $x_{1..n}$ and a performance metric v. The goal is to build a model based on the entity data which can maximize the function of the performance metric. For example, a task might be spam detection, where e-mails are automatically classified as ham or spam. Here the entities are e-mails, while the performance metric v is the number of correctly classified e-mails on a test set.

There are two main types of machine learning tasks. Firstly *classification* is a task when the machine learning model predicates a label to the given instance from a set of pre-defined classes (e.g. to classify e-mails as spam or not). Secondly, *regression* forecasts a real value within an interval (like the price of a house) for a given test instance. In this thesis, we shall just deal with classification tasks. Here we will predict what multiword expression class the given unit belongs to if any.

The level of supervision in machine learning refers to the availability of manually labelled data. *Supervised* learning involves learning a target function from training examples of its inputs. *Unsupervised* learning attempts to learn patterns and associations from a set of objects that do not have attached class labels. *Semi-supervised* learning learns from a combination of labeled and unlabeled examples. However, in many cases there is only a limited set of annotated data available on one domain, and sufficient gold standard data on another domain. In this case we can focus on the portability of models trained on a different domain or we can apply *domain adaptation* techniques to reduce the gap between the domains. Domain adaptation is especially useful when there is only a limited amount of annotated data available for one domain but there are plenty of data for the another domain. Using the do-

main with a lot of annotated data as the source domain and a domain with limited data as the target domain, domain adaptation techniques can successfully contribute to the learning of a model for the target domain (see e.g. Chapter 7).

4.3 Support Vector Machine

One of the most widely applied machine learning classification methods is Support Vector Machines (SVMs) (Cortes and Vapnik, 1995). It shows promising empirical results in many practical problems like handwritten digits recognition or text classification. SVM is well suited to work with high dimensional data as it avoids the high dimension bias problem.

The training set of SVMs consists of feature vectors with associated labels $(\vec{x_i}, y_i \in \{-1, 1\})$. As each vector has the same number of components, we can consider the vectors as one vector of the same n-dimensional vector space.

SVM is based on the main idea of selecting a directed hyperplane (decision hyperplane) that separates the vector space (between the positive and negative classes), while maximizing the smallest margin of the hyperplane. The margin consists of two separating hyperplanes on the two sides of the decision hyperplane with the same normal vector and orientation as the decision hyperplane. They are given by the following:

$$H_1 : (\vec{x_i}\vec{w}) + b = 1, \text{ where } y_i = 1$$
$$H_2 : (\vec{x_i}\vec{w}) + b = 1, \text{ where } y_i = -1$$

The distance H_1 is $\frac{|1-b|}{\|w\|}$ while the the distance H_2 is $\frac{|-1-b|}{\|w\|}$, which makes $\frac{2}{\|w\|}$ wide margin. The decision boundary is optimal when the $\|w\|^2$ value is maximal. We call $\vec{x_i}$ as *support vector*, if we eliminate $\vec{x_i}$ from the model, the separating hyperplanes are changed. In the prediction phase, the label of the feature vector is determined by which side of the directed hyperplane it is situated. In this thesis, we used libSvm[1] and the Weka **SMO** implementation.

4.4 Decision Trees

Decision tree learning is a method for approximating discrete-valued target functions, in which the learned function is represented by a decision tree.

Decision trees classify instances by sorting them down the tree from the root to some leaf node, which provides the classification of the instance. Each node in the tree specifies a test of some attribute of the instance, and each branch descending from that node corresponds to one of the possible values for this attribute. An instance is classified by starting at the root node of the tree, testing the attribute specified by this node, then moving down the tree branch corresponding to the value of the attribute in the given example. This process is then repeated for the subtree rooted at the new node.

The basic algorithm called ID3 learns decision trees by constructing them top-down, beginning with the question "which attribute should be tested at the root of the tree"?

[1] goo.gl/2K7Kl4

We trained the J48 classifier of the WEKA package (Hall et al., 2009), which implements the decision trees algorithm C4.5 (Quinlan, 1993), which is based on the ID3 tree learning algorithm. When we built decision trees, they had at least 2 instances per leaf, and we used pruning with subtree raising and a confidence factor of 0.25. We applied decision trees models in Chapter 8.

4.5 Conditional Random Fields

Conditional random fields (CRFs) are a type of discriminative, undirected, probabilistic graphical model for labeling and segmenting structured data, such as sequences, trees and lattices (Lafferty et al., 2001a). The main advantage of CRFs over other sequence labeling methods like Hidden Markov Models (HMMs) is their conditional nature: while HMM carries out a local distribution estimation, in contrast, CRF defines a conditional probability distribution over the whole label sequences given a particular observation sequence. In other words, the CRF model does not work with local probabilities like $p(y_t|x_t)$, where t is the position of x within the sequence, instead, it estimates the conditional probability of the whole sequence:

$$p(y|x) = \frac{1}{Z(x)} exp\{\sum_{t} \sum_{j=1}^{K} \lambda_j f_j(x, y_t, y_{t-1})\}$$

In addition, CRF avoids the label bias problem, as the model optimizes the whole label sequence of conditional probabilities. It creates the connection among the label of time t and subsequent labels $t <$ (non-directional model). The estimation of weights (λ_j) for each feature f_j is carried out by maximizing the conditional log likelihood:

$$\max_{\lambda} \ell(\lambda) = \max \sum_{i=1}^{N} p(y^{(i)}|x^{(i)}),$$

where the training set consists of N observation sequences $x^{(i)}$ and label sequences $y^{(i)}$. Usually, a quasi-Newton method is applied to optimize CRFs.

A major drawback of CRFs is the training time, as it depends quadratically on the number of class labels and linearly on the number of training instances as well as the average sequence length. However, state-of-the-art sequence labeling applications use CRF models, where time consumption is still tolerable. Therefore, we will also employ CRF models to automatically detect of multiword expressions in natural language texts both in English and Hungarian (see Chapters 5, 6 and 7). We trained the Mallet (McCallum, 2002) implementation of a first-order linear chain CRF classifier with the default settings used by Mallet for 200 iterations or until convergence was reached.

4.6 Evaluation metrics

To measure the effectiveness or provide an objective evaluation of the different machine learning approaches, manually annotated test datasets are required . As the test set is not used

during the training phase, it may contains unseen examples for machine learning models. However, the model can automatically predict labels for each instance in the test set that are comparable with the gold standard labels in the original manual annotation.

As we focused on the automatic detection of multiword expressions, we applied a phrase-based evaluation with $F_{\beta=1}$ scores as the evaluation metric. The training dataset was represented in IOB format, where "B-begin" and "I-inside" denote the tokens belonging to multiword expressions and "O-outside" was used for all other tokens.

When all members of multiword elements were labeled correctly and no other neighbouring words were marked as MWEs, it was accepted as *true positive* (*TP*). When there was an MWE entity in the running text, but the system could not correctly recognize it – that is, the system was able to identify an MWE, but got its boundaries wrong or there was an entity but the system failed to identify it – the MWE was treated as a *false negative (FN)*. In the case of *false positives (FP)*, there was no MWE in the text, but the system supposed that there was one.

In order to compute the F_1-scores, we defined precision and recall scores as follows: the precision score measures how precisely a model predicts a target class, in other words, how many of the instances predicted belonging to a given target class are genuine members of that class. Precision is the ratio of correct predictions to the total predicted:

$$Precision = \frac{TP}{TP+FP}$$

Recall measures the ratio of instances of a class that the system actually recognizes as members of the target class in question. Recall measures the ratio of the true positives over the total:

$$Recall = \frac{TP}{TP+FN}$$

A high precision means a low number of *false positives*, while a high recall means a low number of *false negatives*. And the F_1-score was defined as the harmonic mean of the precision and recall:

$$F_1 = 2 * \frac{Precision*Recall}{Precision+Recall}$$

Furthermore, the accuracy is the proportion of true results (both true positives and true negatives) in the total number of instances:

$$accuracy = \frac{truepositives+truenegatives}{truepositives+falsepositives+truenegatives+falsenegatives}$$

4.7 Summary

In this chapter, we introduced the basic machine learning concepts that will be used in our experiments on automatic detection of multiword expressions. We also described Support Vector Machines, Decision Trees and Conditional Random Fields based models. We also presented the evaluation methodology to measure the performance of our systems developed to detect multiword expressions in natural language texts, which will be presented in Chapters 5, 6, 7 and 8.

Chapter 5

English Nominal Compound Detection with Wikipedia-Based Methods

In this chapter, we present the results of our experiments on the automatic detection of English nominal compounds in running texts using Wikipedia-based methods. We propose two different approaches to the task. Here, we present our dictionary lookup method, which relies heavily on Wikipedia and we also investigate how the growth of Wikipedia contributes to the process. In order to automatically identify nominal compounds, we also applied a machine learning-based method using gold standard and silver standard data. Next, we will show how previously identified nominal compounds affect Named Entity Recognition and vice versa, how nominal compound detection is supported by identified named entities. We will argue that previous knowledge of nominal compounds can enhance NER, while previously identified NEs can assist the nominal compound identification process.

5.1 Nominal Compounds

In different languages, nominal compounds can be spelt in different ways. Here, we focus on English nominal compounds, which may consist of one unit, may be hyphenated or may contain spaces (like *headmaster*, *self-control* and *grammar school*). This is in contrast with some other languages like German or Hungarian where nominal compounds form one orthographical unit. They are spelt as one orthographical word in German like *Schuldnerberatungsstelle* (`debt advice`) or they are spelt as one orthographical word or hyphenated in Hungarian like *úszómedence* "swimming pool" or *időjárás-jelentés* "weather forecast". This fact alleviates their identification as one unit, however, a good morphological analyzer is required to identify their parts (Hedlund, 2002). Table 5.1 lists the spelling rules for nominal compounds in these three languages.

Most of the earlier studies (Ramisch et al., 2010a) regard nominal compounds as `noun`–`noun` bigrams pairs. As Wiki50 is a full-covered MWE annotated corpus where each individual occurrence of a nominal compound was marked in running texts, we were able to examine how many of the nominal compounds are `noun`–`noun` bigrams. Table 5.2 shows the distribution of the part of speech (POS) codes of first tokens of nominal compounds

	English	Hungarian	German
spelling as one word	•	•	•
hyphenated	•	•	
spelling as two or more words	•		

Table 5.1: The spelling rules for nominal compounds in three different languages.

Corpus	Nominal Compounds	Noun		Adjective		Other	
		#	%	#	%	#	%
Wiki50	2929	1820	62.14%	1027	35.06%	82	2.8%
BNC	485	416	85.77%	52	10.72%	17	3.5%

Table 5.2: POS code types of the first words of nominal compounds in the Wiki50 corpus and BNC dataset.

provided by the Stanford POS-tagger (Klein and Manning, 2003) for the Wiki50 and BNC corpora. As Table 5.2 shows, in one third of the cases the first token of the nominal compounds was an adjective. Also, Table 5.3 includes some statistics on the length of NCs. A typical example of a two-token NC is *stock car*; one for a three-token long is *microbiological analytical procedure*; and a four-token NC is *amino acid extraction process*. In the Wiki50 corpus 83.37% (2442 occurrences) are two-part NCs, 13.17% (386 occurrences) are three-part NCs and only 3.46% (101 occurrences) are four-or-more part NCs. As for the BNC dataset, there are 436 (89.89%) two-part NCs, 8.25% (40 occurrences) three-part NCs and 1.86% (9 occurrences) four-or-more part NCs.

Based on the above data, we avoid only focusing on `noun–noun` pairs when we automatically detect nominal compounds in running texts and seek to identify longer NCs as well.

There are two basic approaches for detecting nominal compounds, namely identification and extraction (Kim, 2008). In the case of identification, the goal is to identify each nominal compound occurrence in a running text, i.e. to take input sentences such as *'They were in the swimming pool.'* and mark each NC in it. We will follow this approach in the thesis. In the case of nominal compound extraction, the aim is not to mark each NC in the running text, but just to extract nominal compounds from text, so multiple occurrences of the same NC are not taken into account. The extraction process generates a list of NCs extracted from the texts. Hence NC candidates that occur at least once as an NC within the text are treated as NCs,

Corpus	Nominal Compounds	2		3		4≤	
		#	%	#	%	#	%
Wiki50	2929	2442	83.37%	386	13.17%	101	3.46%
BNC dataset	485	436	89.89%	40	8.25%	9	1.86%

Table 5.3: The number of tokens of the nominal compounds, based on their length.

while non-NC uses of the same unit are ignored. This is well illustrated in interpretations of the following sentence:

The lady fed [her dog] [biscuits]. → The dog ate biscuits.
The lady fed [her] [dog biscuits]. → She ate dog biscuits.
The lady fed [her dog biscuits]. → The dog biscuits were fed.

The phrase *dog biscuits* is not a nominal compound in the first sentence, in contrast it is in the other two.

5.2 Related Work

The semantic interpretation of nominal compounds has been a mainstream problem in the past years (Kim and Baldwin, 2006; Nicholson and Baldwin, 2006; Kim and Baldwin, 2008; Kim and Nakov, 2011). The first step in semantic disambiguation is the task of defining what relations exist in nominal compounds (Levi, 1978; Finin, 1980). Also, some studies focus on automatic nominal compound detection from running text. For example, Bonin et al. (2010) use contrastive filtering in extracting multiword terminology (mostly nominal compounds) from scientific, Wikipedia and legal texts: term candidates are ranked according to their belonging to the general language or the sub-language of the domain. Caseli et al. (2010) developed an alignment-based method for extracting multiword expressions from parallel corpora. This method has also been applied to the pediatrics field (Caseli et al., 2009).

The machine-learning based tool `mwetoolkit` is designed to extract MWEs from running texts, as is illustrated by the case of extracting English nominal compounds from the Genia and Europarl corpora and from general texts (Ramisch et al., 2010b; Ramisch et al., 2010c). In contrast to the above approaches, here we focus on identifying all nominal compounds in English running texts.

5.3 Automatic Detection of Nominal Compounds

In this section, we present our dictionary lookup and machine-learning based approach for detecting nominal compounds in running text, which we will elaborate on below. We will also show how the size of an automatically generated silver standard corpus can affect the performance of the machine learning-based method and also show how previously identified NCs can affect NER and vice versa: how NC detection is assisted by NEs identified earlier. To evaluate our models, the Wiki50 corpus and the BNC dataset were applied (see Chapter 3.9).

5.3.1 Wikipedia-based Dictionary Lookup Method for Detecting Nominal Compounds

To identify nominal compounds, we applied a list got from using English Wikipedia. In this case the actual state of the whole English Wikipedia on 1 January 2013 was utilised. Lower-

Method	Recall	Precision	F-score
Match	56.06	38.30	45.51
Merge	59.44	40.77	48.37
POS-patterns	50.60	56.59	53.42
Combined	53.67	59.84	56.59

Table 5.4: Results got from using Wikipedia-based dictionary lookup methods for nominal compounds in terms of recall, precision and F-score. **Match:** dictionary match, **Merge:** merge of two overlapping nominal compounds, **POS-rules:** matching of POS-patterns, **Combined:** the union of Match, Merge and POS-rules.

case n-grams which occurred as links were collected from 9,914,544 Wikipedia articles and the list was automatically filtered so as to delete non-English terms, named entities and non-nominal compounds. The resulting list consisted of 687,574 potential nominal compounds.

We applied three basic methods for NC detection. In the case of the 'Match' method, a nominal compound candidate was only marked when it occurred in the extracted list of n-grams. The second method was applied to nominal compounds that involved the merge of two possible nominal compounds: if `A B` and `B C` both occurred in the list, `A B C` was also accepted as a nominal compound ('Merge'). In other words, if *mobile internet access* occurred in the running text, and the candidate list only contained *mobile internet* and *internet access*, we treated the whole phrase as a nominal compound.

In the case of 'POS-rules', a nominal compound candidate was marked if it occurred in the list and its POS-tag sequence matched one of the typical NC patterns (e.g. `adjective + noun`). POS-tags were determined by the Stanford POS Tagger (Toutanova and Manning, 2000). Afterwards, we 'combined' the results got from using these three methods. So, a nominal compound candidate was marked if any of these three methods yielded the candidate.

Table 5.4 shows the results of applying Wikipedia-based dictionary lookup methods for nominal compound identification on the Wiki50 corpus. We applied recall, precision and F-score as evaluation metrics (see Section 4.6 for details). Here, we observe that the best results can be achieved when the results got from using the three methods are combined.

5.3.2 Machine Learning-based Method for Detecting Nominal Compounds

In order to automatically identify nominal compounds, we also applied a machine learning-based method. The tool uses the MALLET implementations (McCallum, 2002) of the Conditional Random Fields (CRF) classifier (Lafferty et al., 2001b), and the feature set employed was developed on the basis of a general named entity feature set (Szarvas et al., 2006b). Identifying multiword named entities and nominal compounds can be carried out in a similar way as both nominal compounds and multiword named entities consist of more than one word in a sequence. They form one semantic unit and thus, they should be treated as one unit in an

leave-one-out	Recall	Precision	F-score
NC	57.91	69.46	63.16
NC + NE	65.35	72.14	68.58

Table 5.5: Results got from using the leave-one-out approaches in terms of recall, precision and F-score in the Wiki50 corpus. **NC:** our CRF-based approach, **NC + NE:** our CRF with NC features and NEs as additional feature.

NLP system (Nagy T. et al., 2011a; Sag et al., 2002).

Here, our basic named entity feature set was based on the following categories: **orthographical features:** capitalization, word length, bit information about the word form (contains a digit or not, has an uppercase character inside the word, etc.), character level bi/trigrams, suffixes; **dictionaries** of first names, company types, denominators of locations; **frequency information:** frequency of the token, the ratio of the token's capitalized and lowercase occurrences, the ratio of capitalized and sentence beginning frequencies of the token, which was derived from the Gigaword dataset; **shallow linguistic information:** part of speech; **contextual information:** sentence position, trigger words (the most frequent and unambiguous tokens in a window around the word) from the training database and the word between quotes.

This basic named entity feature set was extended with features adapted to nominal compounds. The **dictionaries** were extended with different nominal compound lists. We collected a potential nominal compound list from Wikipedia described in Section 5.3.1 and sorted it according to frequency of the occurrences. The components with different frequencies were included in different dictionaries. In addition, the training and test sets of Task 9 of the SemEval 2010 challenge (Erk and Strapparava, 2010) were used as dictionaries. The shallow linguistic features were extended with the **POS-rules**, so when the POS-tag sequence in the text matched one pattern typical of nominal compounds (e.g. `noun` – `plural noun`), the sequence tags were marked as *true*, otherwise they were marked as *false*. Furthermore, **other entities** were also specified in the sentence like NEs and LVCs, which were also used as features. To identify named entities, the Stanford Named Entity Recognition tool was applied (Finkel et al., 2005) and we looked for LVCs in a similar way to that described in Nagy T. et al. (2011b).

We trained the first-order linear chain CRF classifier with the above-mentioned feature set and evaluated it on the Wiki50 corpus in a 10-fold cross-validation setting at the sentence level. We trained CRF models with L1 regularization and the default settings in Mallet for 200 iterations or until convergence was attained. The *NC* and *NC + NE* rows in Table 5.5 show the results of applying this method on the Wiki50 corpus. These results tell us that a knowledge of named entities is helpful in the identification of nominal compounds in English running text.

These results may be related to the fact that multiword NEs and nominal compounds are similar from a linguistic point of view, as discussed earlier. Moreover, in some cases even for humans, it is not easy to determine whether a given sequence of words is an NE or

Approach	Recall	Precision	F-score	Recall	Precision	F-score
	Identification evaluation scheme			Extraction evaluation scheme		
mwetoolkit	–	–	–	12.41	38.32	18.75
Dictionary Lookup	52.47	59.45	55.75	50.10	60.46	54.81
CRF	44.38	58.42	50.44	43.69	60.10	50.60
CRF + SF	53.39	56.66	54.98	52.94	57.57	55.15

Table 5.6: Results got from using different methods for nominal compounds in terms of recall, precision and F-score in the Wiki50 corpus. **mwetoolkit:** the mwetoolkit system, **Dictionary Lookup:** Wikipedia-based dictionary lookup method, **CRF:** our CRF model trained on an automatically generated database, **CRF + SF:** our CRF model trained on sentences with at least one NC label.

NC (capitalized names of positions such as *Prime Minister* or taxonomic names, e.g. *Torrey Pine*). In the test databases, no unit was annotated as an NE and NC at the same time, thus it was necessary to disambiguate cases which might be labeled by both the NC and the NE systems. After fixing the label of such cases, disambiguity was eliminated, that is, the training data sets were less noisy, which leads to better overall results.

5.3.3 Training on a Silver Standard Dataset

We experimented with training on the automatically generated silver standard corpus, hence we were able to use the Wiki50 corpus for testing only. We applied our dictionary-based method (see Section 5.3.1) to automatically generate a silver standard dataset. It consisted of 5,000 randomly selected Wikipedia pages that do not contain lists, tables or other structured texts. These documents were not manually annotated, so here the dictionary-based NC labeling approach was treated as a silver standard. The resulting dataset was much bigger than the manually annotated corpora available, but the annotation was less reliable. Here, we just wanted to exploit the benefits of having a big training data with less accurate annotation.

As we also wished to see how previously identified named entities affected nominal compound detection, the CRF model was trained on the automatically generated silver standard dataset with the feature set presented in Section 5.3.2 without an NE feature. The results can be seen in the *CRF* row of Table 5.6.

The database included many sentences without any labeled nominal compounds hence negative examples were overrepresented. Therefore, we thought it necessary to filter the sentences: only those with at least one nominal compound label were retained in the database. As we see in the *CRF + SF* rows in Tables 5.6 and 5.7, we were able to build a better model when this filtering methodology was applied.

We also investigated what results could be achieved using the Wikipedia-based approaches on another corpus. This is why we evaluated our methods on the BNC dataset as well. In Table 5.7 it is clear that our approaches achieve poorer results on the BNC dataset than those got on the Wiki50 corpus. This is probably due to the fact that our approaches rely heavily

Approach	Recall	Precision	F-score	Recall	Precision	F-score
	Identification evaluation scheme			Extraction evaluation scheme		
mwetoolkit	–	–	–	10.22	18.84	13.26
dictionary lookup	30.39	37.13	33.42	31.31	42.25	35.97
CRF	27.27	40.49	32.59	30.44	42.20	35.37
CRF + SF	34.91	39.48	37.06	39.11	41.33	40.19

Table 5.7: Results of different methods for nominal compounds in terms of recall precision and F-score in the BNC dataset. **mwetoolkit:** the mwetoolkit system, **dictionary lookup:** Wikipedia-based dictionary lookup method, **CRF:** our CRF model trained on automatically generated database, **CRF + SF:** our CRF model trained on sentences with at least one NC label.

on the Wikipedia and there are differences between the two corpora. As mentioned in the BNC paper (Nicholson and Baldwin, 2008), they annotated sequences of two nouns. Because of this, the method of merging overlapping nominal compounds could not be applied here. However, in the BNC dataset we found 40 three-part, and 9 longer nominal compounds annotated in the data. On the other hand, some of the errors were related to annotation errors, like marking NCs that contain a proper noun (e.g. *Belfast primary school headmaster*), as simple NCs instead of proper nouns (as they should be according to the guidelines). These differences might be responsible for the weaker performance of our methods on the BNC dataset.

In order to compare our methodology with other systems, we wanted to see what results the other systems could achieve using our corpora. But we found only one available system for English nominal compound detection. This is the mwetoolkit system (Ramisch et al., 2010a), a language-independent tool developed for collecting MWEs from texts (which is able to identify nominal compounds). We evaluated it on both corpora as well. This system also relies heavily on POS tag features, hence we completed the mwetoolkit POS tag rules with our POS rules. However, the mwetoolkit basically does not really mark MWEs in the running text, but just extracts nominal compounds from the text, so multiple occurrences of the same MWE are not taken into account. Therefore, in order to compare the results of our approaches with those of mwetoolkit, it was necessary to assess our methods in a similar way to the evaluation scheme used in mwetoolkit. The results of mwetoolkit and our methods on the Wikipedia corpus can be seen on the right hand side in Tables 5.6 and 5.13 and the BNC dataset on the right hand side in Tables 5.7 and 5.14. As the tables show, with this evaluation method we get better F-scores. This is due to the fact that if a particular phrase occurs several times in the text and we cannot identify it, it is marked as only one error in this evaluation, and in the other evaluation, each occurrence of the same nominal compound must be identified and multiple occurrences are marked as multiple errors. The right hand side of Tables 5.6 and 5.7 shows that we were able to achieve considerably better results using our method than that got using mwetoolkit.

To perform an error analysis, we examined the length of nominal compounds in the

	LOO			Silver standard			Dictionary lookup		
	Recall	**Precision**	**F-score**	**Recall**	**Precision**	**F-score**	**Recall**	**Precision**	**F-score**
2	69.12	79.62	74.00	64.86	60.14	62.41	61.14	64.66	62.85
3	52.33	62.93	57.14	29.02	47.86	36.13	30.05	49.79	37.48
4≤	24.73	45.10	31.94	8.60	40.00	14.16	6.45	75.00	11.88
All	65.35	72.14	68.58	56.57	55.57	56.06	53.67	59.84	56.59

Table 5.8: Results got from using different methods for nominal compounds in terms of recall, precision and F-score in the Wiki50 corpus. **LOO:** CRF model evaluated in the leave-one-document-out scheme. **Silver standard:** CRF model trained on the automatically generated silver standard dataset. **Dictionary lookup:** Wikipedia-based dictionary lookup method.

	LOO			Silver standard			Dictionary lookup		
	Recall	**Precision**	**F-score**	**Recall**	**Precision**	**F-score**	**Recall**	**Precision**	**F-score**
2	39.68	39.50	39.59	40.60	45.04	42.70	33.49	45.06	38.42
3	20.00	21.62	20.78	20.00	22.86	21.33	17.50	17.95	17.72
4≤	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
All	34.91	39.48	37.06	38.02	41.53	39.70	31.40	40.75	35.47

Table 5.9: Results got from using different methods for nominal compounds in terms of recall, precision, and F-score in the BNC dataset. **LOO:** CRF model evaluated in 10-fold cross-validation setting at the sentence level. **Silver standard:** CRF model trained on the automatically generated silver standard dataset. **Dictionary lookup:** Wikipedia-based dictionary lookup method.

corpora. Tables 5.8 and 5.9 show that all the methods achieved their best results on the two-part nominal compounds. Longer nominal compounds yielded worse results for each method and corpus used.

Tables 5.8 and 5.9 also give the results from using the different approaches for the Wiki50 and BNC datasets. Table 5.8 reveals that on the Wiki50 corpus, the CRF model evaluated with the leave-one-document-out scheme yielded the best results with an F-score of 68.58. The CRF model trained on the automatically generated silver standard dataset and the Wikipedia-based dictionary lookup method achieved roughly the same F-score on the Wiki50 corpus, but produced different recall and precision scores. The machine learning-based method yielded a higher recall on the one hand and a lower precision on the other. As we can see in Table 5.9, the CRF model trained on the automatically generated silver standard dataset yielded an F-score that was 2.64 higher than with 10-fold cross-validation and 4.23 higher on the BNC dataset than that using the dictionary lookup method. As the BNC dataset is relatively small, the machine learning-based approach could not train an accurate model in the 10-fold cross-validation setting. However, the machine learning-based method can generalize the characteristics of nominal compounds when it was trained on the out-domain and the larger automatically generated dataset and build a better model.

We also carried out an error analysis and examined the nature of English nominal compounds. We found that the majority of nominal compounds are two-part and the investigated

approaches performed well on the two-part compounds in contrast to longer compounds, which is probably due to the fact that our automatically labeled examples contained fewer instances of longer compounds.

5.3.4 The Expansion of the Training Set Size

The CRF model was trained on the silver standard dataset with the feature set given in Section 5.3.2. First, we investigated how the size of the automatically labeled silver standard training set influenced the performance of CRF. We analyzed the results when the training set only consisted of 10 Wikipedia pages. After, we gradually increased the automatically labeled training set by adding a random selection of Wikipedia pages.

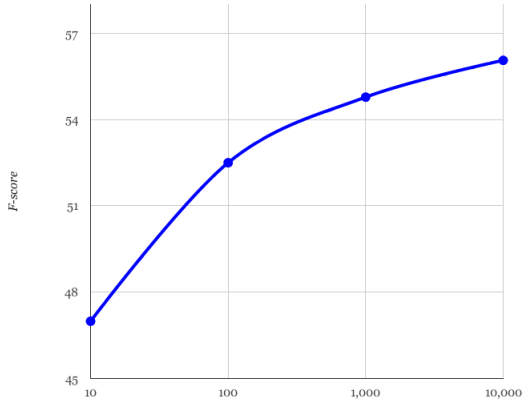

Figure 5.1: Results got from using the machine learning approach as a function of the automatically generated silver standard training set size (the number of Wikipedia pages).

As Figure 5.1 shows, with an increased training set the machine learning approach was able to produce better results, but the improvement was smaller. The method produced an F-score of 46.69 when the training set just consisted of 10 Wikipedia pages and an F-score of 56.06 when it was constructed from 10,000 Wikipedia pages.

As we used randomly selected Wikipedia pages to train our CRF model, we investigated how the random selection affected the performance. We automatically generated ten different training sets. Here, one set consisted of ten thousand randomly selected Wikipedia pages, where dictionary based labeling was used as the silver standard and a CRF model was trained with the feature set described in Section 5.3.2 . Table 5.10 lists the results got from using ten different CRF model results, trained on ten different automatically generated datasets. The average F-score of ten runs was 55.99 and the standard deviation was 0.3237. The method proved to be sufficiently robust as the standard variation was relatively small.

We investigated the method presented in Section 5.3.1 from the beginning of Wikipedia and to see how the size of Wikipedia influenced the results. This was why we collected items

	Recall	Precision	F-score
1	57.02	55.21	56.1
2	56.74	55.38	56.05
3	57.26	55.73	56.48
4	56.64	55.02	55.82
5	57.46	55.25	56.33
6	56.88	55.61	56.24
7	56.98	55.03	55.99
8	56.2	54.94	55.56
9	57.08	53.73	55.36
10	56.85	55.04	55.93
avg.:	56.91	55.1	55.99

Table 5.10: Machine learning results for Wiki50 obtained on different samples of automatically generated silver standard training sets in terms of recall, precision, and F-score.

for the nominal compound list presented in Section 5.3.1 from the actual state of Wikipedia at the beginning of each year. English Wikipedia was launched in 2001, so the first list was collected from the state of 1 January 2002, while the last one was 1 January 2013.

Table 5.11 shows results got from using the Wikipedia-based dictionary lookup approach, the number of Wikipedia pages and the size of the collected lists, as a function of years and the actual state of Wikipedia.

After the first year, English Wikipedia only consisted of 13,200 pages, and we were able to extract 5,892 potential nominal compounds from the links. The Wikipedia-based dictionary lookup method yielded an F-score of 9.52 on the Wiki50 corpus. While at the beginning of 2013 the English Wikipedia consisted of 9,914,544 pages, the potential NC list contained 687,574 elements and using this approach we got an F-score of 56.59. As Table 5.11 shows, based on the growth of Wikipedia, the method was able to produce better results, but the rate of improvement was negligible after 2007. Furthermore, in 2013 the dictionary-based method yielded an F-score that was 0.15 lower than that in 2012. Figure 5.2 also shows how the growth of Wikipedia contributes to the results got by using the Wikipedia-based dictionary lookup method.

5.3.5 Named Entity Recognition with Nominal Compounds

In this section we will examine how previously identified nominal compounds affect Named Entity Recognition. Therefore, we investigated the usefulness of nominal compounds in Named Entity Recognition. We used the Wiki50 corpus to train CRF classification models with the basic named entity feature set (Szarvas et al., 2006b) with the nominal compound feature added. The model was evaluated on the Wiki50 corpus in a leave-one-document-out scheme. The results of this approach are shown in the *NE + NC* row of Table 5.12. Comparing these results with those of the *NE* method (when the CRF was trained without the nominal compound feature), we see that the use of nominal compounds assists the process

Year	WikiPages	NC list	Recall	Precision	F-score	Difference
2002	13,200	5,892	5.12	68.42	9.52	-
2003	124,229	25,431	16.22	59.05	25.45	+15.93
2004	271,160	58,696	24.99	71.69	37.06	+11.61
2005	752,239	120,028	33.81	69.57	45.50	+8.44
2006	1,611,876	211,802	40.11	66.20	49.96	+4.46
2007	2,988,703	322,918	44.42	64.15	52.49	+2.53
2008	4,432,034	405,635	46.91	63.35	53.90	+1.41
2009	5,281,708	459,544	48.51	62.82	54.74	+0.84
2010	6,009,776	511,303	49.33	62.45	55.12	+0.38
2011	7,167,621	567,288	50.69	62.66	56.04	+0.92
2012	9,007,810	640,879	53.36	60.58	56.74	+0.7
2013	9,914,544	687,574	53.67	59.84	56.59	-0.15

Table 5.11: The results got from applying the Wikipedia-based dictionary lookup method, as a function of the size of the Wikipedia, measured in terms of recall, precision, and F-score. **WikiPages:** the number of Wikipedia pages. **NC list:** the size of the lists collected from the Wikipedia links.

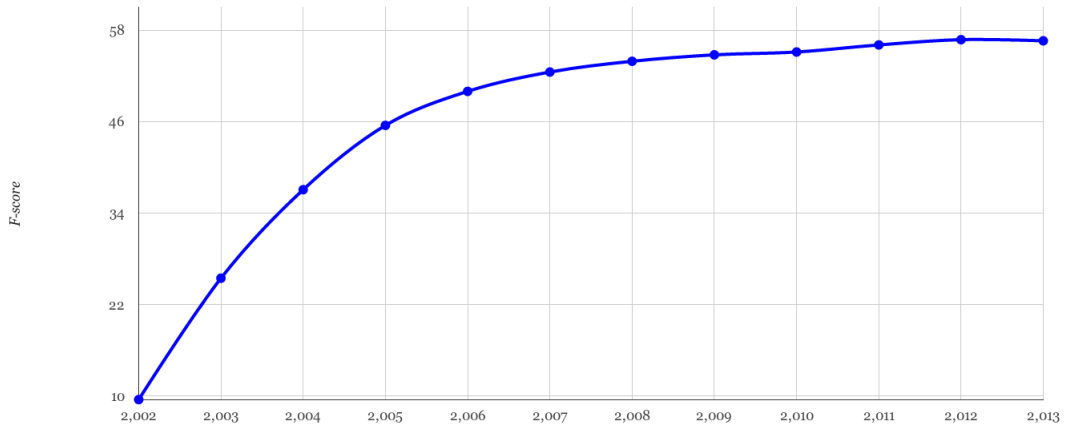

Figure 5.2: Results got from applying the Wikipedia-based dictionary lookup method, as a function of the size of Wikipedia, measured in terms of F-score.

leave-one-out	Recall	Precision	F-score
NER with basic feature set	84.79	85.17	84.98
NER with basic and NC features	86.03	86.37	86.20

Table 5.12: Named Entity Recognition results of applying the leave-one-out approaches on the Wiki50 corpus in terms of recall, precision and F-score.

of NER.

First, the Stanford NER model trained on the English business short news (Finkel et al., 2005) was used to identify NEs. However, we assumed that a model trained on Wikipedia could more effectively identify NEs in Wikipedia (the same domain). Therefore, we merged the four NE classes marked in Wiki50 into one NE class in order to train the CRF with the common feature set. Our results are shown in the *NE* row of Table 5.12.

5.3.6 Detecting Nominal Compounds with Named Entities

In Section 5.3.5 we investigated how NER was supported by identified nominal compounds. Here, we examine how NER can help the nominal compound detection.

The CRF model for nominal compound detection decribed in 5.3.2 was also trained on the silver standard dataset with the basic feature set extended with the information that a token is a named entity or not. The *NC + NE* row of Table 5.5 shows that this feature proved effective in the leave-one-document-out scheme, so we used it in the automatically generated silver standard database as well. As shown in the *CRF + NE* row of Table 5.13, the CRF model trained on the automatic training set got better results with this feature than those for the original *CRF*.

As sentence filtering yielded better results (see Section 5.3.3), we decided to use this approach. The *CRF + OwnNE + SF* row in Table 5.13 represents results achieved when the NEs that were identified using the entire Wiki50 corpus as the training dataset functioned as a feature. Although the *CRF + NE + SF* model (when NEs were identified by the Stanford model) did not produce better results than those got using the *CRF + SF* model, our Wikipedia-based NE CRF model used to identify NEs in the automatically generated training dataset (*CRF + OwnNE + SF*) yielded a better F-score than that for *CRF + SF*, which means that NE is a good feature for the identification of nominal compounds.

Some linguistic units may behave like nominal compounds as well as named entities (e.g. *City Hall*). Nevertheless, we assumed that a term could occur either as an NE or a NC. Hence, if the dictionary lookup method marked a particular word as a nominal compound and the NE model also marked it as an NE, we had to decide which mark to delete. The `CRF + OwnNELeft + SF` row in Table 5.13 shows results we got when the NE labeling was selected as a feature and the standard nominal compound notation was removed, while the row *CRF + NCLeft + SF* represents the case where the NE feature was deleted, and the standard nominal compound notation was kept. Also, in Table 5.14 we see similar trends got for the BNC dataset.

Approach	Recall	Precision	F-score	Recall	Precision	F-score
	Identification evaluation scheme			Extraction evaluation scheme		
mwetoolkit	-	-	-	12.41	38.32	18.75
CRF	44.38	58.42	50.44	43.69	60.10	50.60
CRF + NE	45.81	58.37	51.33	45.16	59.84	51.48
CRF + NE + SF	53.12	55.89	54.47	52.72	57.26	54.90
CRF + OwnNE + SF	53.29	57.60	55.36	52.84	59.8	56.13
CRF + OwnNELeft + SF	53.44	57.60	55.44	53.32	59.81	56.38
CRF + NCLeft + SF	53.53	58.74	56.02	53.01	59.67	56.14

Table 5.13: Results obtained for different methods for nominal compounds in terms of recall, precision and F-score on the Wiki50 corpus. **mwetoolkit:** the mwetoolkit system, **CRF:** our CRF model trained on an automatically generated database, **SF:** sentences without any NC label filtered, **NE:** NEs marked by the Stanford NER used as a feature, **OwnNE:** NEs marked by our CRF model (trained on Wikipedia) used as a feature, **OwnNELeft:** the NE labeling selected as a feature, with the standard nominal compound label deleted, **NCLeft:** the standard nominal compound label selected as a feature, with named entity label deleted.

As Tables 5.13 and 5.14 show, using the fact that a token is a named entity or not along with sentence filtering effectively improved the machine learning-based trained model on an automatically generated silver standard dataset.

5.4 Discussion

Here, we investigated dictionary and machine learning-based methods for identifying nominal compounds in English texts in two different corpora. These approaches made intensive use of Wikipedia data. The dictionary-based approach applied a list automatically collected from Wikipedia. Due to the dynamic expansion of Wikipedia, the dictionary-based method was able to extract bigger potential nominal compound lists from Wikipedia links and it achieved better recall scores with each year. At the same time, while the automatically extracted list was noisy, the precision score continuously decreased over the years. However, we found that the dynamic expansion of Wikipedia had a beneficial effect on the recall score, so to improve the precision score we should define restricted rules for labelling nominal compounds.

In order to automatically identify nominal compounds, we also applied a machine learning-based method. As we treated as nominal compounds in a similar way as named entities, we employed a basic named entity feature set, extended with features adapted to nominal compounds. We also looked at the effectiveness of the machine learning-based method when it was trained on an automatically generated silver standard corpus and we demonstrated that this approach can also provide acceptable results.

When we compare the efficiency of the different methods we found that the machine learning-based method yielded the highest F-score value on the Wiki50 corpus as here we

Approach	Recall	Precision	F-score	Recall	Precision	F-score
	Identification evaluation scheme			Extraction evaluation scheme		
mwetoolkit	-	-	-	10.22	18.84	13.26
CRF	27.27	40.49	32.59	30.44	42.20	35.37
CRF + NE	27.27	38.70	31.99	30.44	40.88	34.89
CRF + NE + SF	31.97	40.73	35.83	38.64	43.65	40.99
CRF + OwnNE + SF	36.78	36.10	36.43	41.22	37.93	39.50
CRF + NELeft	40.28	39.35	39.81	44.68	40.29	42.37
CRF + NCLeft	36.57	40.60	38.48	40.98	42.68	41.81

Table 5.14: Results obtained for different methods for nominal compounds in terms of recall, precision and F-score on the BNC dataset. **mwetoolkit:** the mwetoolkit system, **CRF:** our CRF model trained on an automatically generated database, **SF:** sentences without any NC label filtered, **NE:** NEs marked by the Stanford NER used as feature, **OwnNE:** NEs marked by our CRF model (trained on Wikipedia) used as a feature, **OwnNELeft:** the NE labeling selected as a feature, with the standard nominal compound notation deleted, **NCLeft:** the standard nominal compound label selected as a feature, with named entity label deleted.

applied a supervised model. However, the machine learning-based model trained on the automatically generated silver standard dataset and the Wikipedia-based dictionary lookup method achieved roughly the same F-score on the Wiki50 corpus, but produced different recall and precision scores. The machine learning-based method achieved a higher recall on the one hand and a lower precision on the other. On the BNC dataset, the machine learning-based model trained on the automatically generated silver standard dataset outperformed the supervised model and the dictionary lookup method. The machine learning-based approach could not create an accurate model in the 10-fold cross-validation setting as the BNC dataset is relatively small. However, the machine learning-based approach can generalize the characteristics of nominal compounds, when we trained our method on the out-domain and the larger, automatically generated silver standard dataset.

We also investigated the efficiency of dictionary-based method from the beginning of Wikipedia and to see how the size of Wikipedia influenced the results. Due to the dynamic expansion of Wikipedia, the dictionary-based method was able to extract bigger potential nominal compound lists from Wikipedia links and it achieved better recall scores with each year. At the same time, while the automatically extracted list was noisy, the precision score continuously decreased over the years, while the F-score improvement continuously decreased. However, we found that the dynamic expansion of Wikipedia had a beneficial effect on the recall score, so to improve the precision score we should define restricted rules for labelling nominal compounds.

We also examined how the training set size influenced the performance of this machine learning-based approach and we found that with an increased training set the machine learning method was able to produce better results, but the improvement was smaller. We also wanted to see how the random selection of Wikipedia pages affected the performance of the

machine learning-based approach. The method proved to be sufficiently robust as the standard variation was relatively small, when we trained the model on automatically generated ten different training sets and evaluated on the Wiki50 corpus.

Our results demonstrate that previously known nominal compounds are beneficial in NER and identified NEs enhance NC detection. This may be related to the fact that multiword NEs and nominal compounds are similar from a linguistic point of view, moreover, in some cases, it is not easy to determine even for humans whether a given sequence of words is a NE or a MWE (capitalized names of positions such as *Prime Minister* or taxonomic names, e.g. *Torrey Pine*).

5.5 Summary of thesis results

The main findings presented in this chapter are the following:

- We developed a Wikipedia-based dictionary lookup method to automatically detect NCs on English raw texts.
- We implemented a supervised machine learning-based model to automatically detect NCs. The model was trained on the Wiki50 corpus.
- We trained a machine learning-based model on the automatically generated silver standard dataset.
- We evaluated our methods on the Wiki50 corpus and the BNC dataset. The supervised model achieved the highest F-score value on the Wiki50 corpus, while the model trained on the silver standard dataset was the most successful on the BNC dataset.
- We also demonstrated how the size of an automatically generated silver standard corpus could affect the performance of our machine learning-based method. The results we obtained reveal that the bigger the dataset, the better the performance will be.
- We presented the results of our experiments on how the size of Wikipedia could improve the performance of our Wikipedia-based dictionary lookup method for detecting nominal compounds. We found that the growth of Wikipedia improved the performance, especially the recall score, but the rate of improvement diminished over time.
- We demonstrated the usefulness of NCs in Named Entity Recognition and vice versa, and presented how NC detection was supported by identified named entities. The results indicated that the knowledge of named entities is useful in the NC identification process and known nominal compounds can assist Named Entity Recognition.

In Vincze et al. (2011b), the Wiki50 corpus was presented along with the primary results got by using dictionary lookup methods. The author developed the dictionary-based method to automatically detect nominal compounds. One of the co-authors annotated the corpus and provided the linguistic background.

In Vincze et al. (2011a), nominal compounds are identified in running text with rule-based methods. The author developed the dictionary lookup and rule-based methods for the automatic detection of nominal compounds and light verb constructions, and compared the effect of the different features. The co-authors were responsible for linguistic analysis of the data.

In Nagy T. et al. (2011a), nominal compounds and named entities were identified, and we investigated how they can contribute to keyphrase extraction, furthermore we also examined how previously identified nominal compounds affected Named Entity Recognition and vice versa, how nominal compound detection is supported by identified named entities. The author implemented the machine learning-based nominal compound detector and examined the effectiveness of the previously known named entities and nominal compounds on Named Entity Recognition and nominal compound detection, respectively. The co-authors were responsible for the linguistic analysis of nominal compounds and named entities and keyphrase extraction experiments.

In Nagy and Vincze (2013), Wikipedia-based methods were presented for the automatic detection of nominal compounds. The author investigated how the size of an automatically generated silver standard corpus can affect the performance of the machine learning-based method, as well as how the growth of the Wikipedia added to the performance of the dictionary lookup method. The co-author was responsible for the linguistic background.

Chapter 6

Named Entity Recognition

Named Entity Recognition is a key part of information extraction systems because named entities (like person names) are the main building blocks of relations and events. Moreover, the semantic classification (e.g. organisation, person name or location) of the entities is a more challenging problem than simple recognition and for this, one often needs information based on the context of the mentions. We consider named entities similar to nominal compounds as NEs form one semantic unit and can consist of more than one word and they function as a noun. Figure 6.1 shows the relationship among named entities, nominal compounds and multiword expressions. A similar approach could be applied for their recognition as in the case of nominal compounds described in Chapter 5. In this chapter we will focus on Web Mining-based named entity recognition problems.

6.1 Named Entity Recognition for English and Hungarian

One of the key task of natural language processing systems is that of identifying named entities and classify the proper semantic class (person, organization, location names) in documents, as they usually play an important role. Named Entity Recognition was a task assigned within the framework of the Message Understanding Conference MUC-7 (Chinchor, 1998). Participants had to identify personal names, geographical names, organisations, and other names related to time, quantity, and descriptive terms. In 2003, The Conference on Computational Natural Language Learning (CoNNL) (Sang and Meulder, 2003) was announced by the open tournaments. The aim was to construct a Named Entity Recognition model that could handle English and German texts. However, there are some important differences between the CoNLL task definition and the MUC approach. The most important difference is that CoNNL just considers whole phrases classified correctly (which is more suitable for real-world applications). The majority of named entities are multiword named entities (see Tables 6.2 and 6.7). Therefore, the NE phrases can be treated as sequences. One of the most successful and most widely used approaches for sequence labeling is the Conditional Random Fields approach (see Section 4.5). The best performing systems on the CoNLL task applied a CRF approach and gave an accuracy score of 85-89% for English (Ratinov and Roth, 2009).

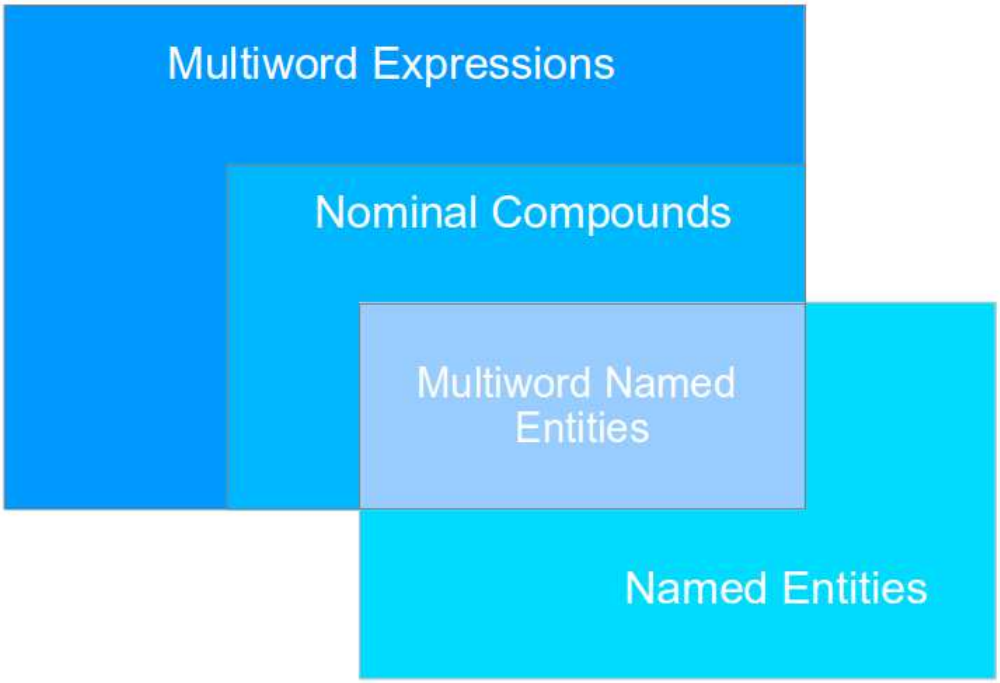

Figure 6.1: Connection among named entities, nominal compounds and multiword expressions.

For the Hungarian language, there exist rule-based (Simon, 2013) and machine learning Named Entity Recognition tools (Farkas et al., 2006; Varga and Simon, 2007). The statistical systems are based on the Hungarian Named Entity Corpus of Business Newswire Texts (Farkas et al., 2006). Our system contains Conditional Random Fields-based NER modules, which were applied on web content texts.

6.2 Named Entity Recognition Problems in Web Mining

Here, we present three different Named Entity Recognition problems from the field of web mining, namely Researcher Affiliation Extraction from English websites, Person Attribute Extraction from English webpages and Company Contact Information Extraction from Hungarian pages, which make use of Named Entity Recognition.

In the case of the researcher affiliation extraction problem, the goal is to provide a deeper insight into a research field or into the personal connections among fields by analysing relationships among researchers. The existing studies use the co-authorship (Newman, 2001; Barabasi et al., 2001) or/and the citation information (Goodrum et al., 2001; Teufel et al., 2006) – generally by constructing a graph with nodes representing researchers – as the basis for their investigations. Apart from publication-related relationships – which are presented in structured scholarly datasets –, useful scientific social information can be gathered from

the WWW. Take, for instance, the homepage of researchers where they summarise their *topic of interest*, list of *supervisors* and *students*, *nationality*, *age*, *memberships* and so on. This type of information can be recognised by using NER tools.

Yet another example of a web-mining-based NER application is Person Attribute Extraction. According to the person name disambiguation task in the Web People Search Challenge the person names are among the most frequently searched items in web search engines (Spink et al., 2004). During the evaluation of the first WePS campaign (Artiles et al., 2007), the organizers realised that the person-related information could be useful for disambiguation. Hence, the organisers defined a new independent attribute extraction subtask in the second WePS (Artiles et al., 2009). The subtask involved extracting the values of those attributes as accurately as possible from webpages. The third WePS shared task (Artiles et al., 2010) introduced a novel subtask which sought to mine attributes for persons, i.e. rather than recognizing attributes in webpages, the task was to assign them to people (the clusters of pages belonging to each given person).

As large amounts of useful information are usually available on the internet about companies, their automatic collection might be required. In the case of Extraction of Company Contact Information, our goal is to collect the names and the addresses of companies from their webpages. As the addresses and the names of companies generally consist of more than one word, we can also treat this type of data as multiword named entities, so NER approaches can be also used here.

6.3 General Architecture for Named Entity Recognition in Webpages

General NLP tools have been developed for processing well-formatted texts. Since webpages usually contain several noisy and misleading elements (such as menu elements and ads), these can seriously inhibit the proper functioning of NLP tools, so we applied some methods to normalise the content of webpages to automatically detect named entities. In the first step, we focus on the raw textual parts of the webpages, as we found that most of the useful information is available in natural text format in webpages. Therefore, we automatically detected the relevant sections from the webpages. Then, named entities are automatically detected by machine learning-based models. Lastly, we validated candidate named entities with application-specific rule-based methods.

6.3.1 Paragraph Extraction

First, we investigated by hand where the useful information was available on the webpages. We used the Researcher Affiliation Corpus (see Section 3.4), where the researcher information was manually annotated. We found that the affiliation information was often present on webpages in an itemised or natural text format. Statistically, 47% of the pages contained affiliation information exclusively in a textual form, 24% were exclusively in an itemised form and 29% were hybrid. The information extracted from these two different formats required

different methods. Therefore, we decided to just focus on the natural language-written part of websites and tables, and we discarded a lot of noisy and misleading elements. In order to identify textual paragraphs, we applied the Stanford POS tagger for each section of the DOM tree of the HTML files. We assumed that one piece of text was a textual paragraph if it was longer than 60 characters and it contained more than one verb. Needless to say, this rule is far from perfect (paragraphs describing publication and longer items of lists are still present), but it seems to be a reasonable one as it extracts textual paragraphs even from "hybrid" pages.

6.3.2 Paragraph Filtering

We applied our paragraph extraction method and using in the Researcher Affiliation Corpus we got 86,735 paragraphs in the 5,282 downloaded pages and used them in our experiments in a raw text format (the HTML tags were removed), and 187,032 paragraphs in 5,122 pages in the English Name Disambiguation Test Corpus. However, we discovered that only a small portion of the textual paragraphs extracted contained useful information. To handle this problem, we developed attribute-specific relevant section selection modules for the attributes listed in Table 6.2. Our filtering method gathered the paragraphs containing the searched item (`positive paragraphs`). To solve this task, we calculated the `P(word|positive)` conditional probabilities and then the best words based on this measure (e.g. *university*, *institute* and *professor* in the case of the researcher affiliation identification task) formed the so-called positive wordset. The paragraphs that did not contain any word present in the positive wordset were removed. Note that standard positive and negative sample-based classifications are not applicable here as the non-positive paragraphs may contain these indicative words, but in an irrelevant context or with some connection to people outside of our scope of interest. Our 1-DNF hypothesis described above uses just positive examples and it was inspired by Yu et al. (2002). We applied this paragraph-filtering method on the researcher affiliation and person attribute extraction task as well. After performing this procedure on the Researcher Affiliation corpus, we kept 14,686 paragraphs (from the full set of 86,735), but we did not leave out any annotated text. Hence the information extraction module could then work with a smaller and less noisy dataset. Table 6.1 summarises the size-related statistics for a part of this textual corpus which contains affiliation information (these paragraphs contain manually labelled information).

As attributes are not manually annotated in the English Name Disambiguation Test Corpus, we could not calculate the efficiency of paragraph filtering for this corpus. However, the paragraph filtering method was applied to find the *occupation*, *affiliation*, *award* and *school* attributes.

Data	**#**
researchers	59
pages	103
paragraph	151
sentences	181

Table 6.1: The size of the textual corpus which contains affiliation information.

6.4 Researcher Affiliation Extraction from Homepages

In the researcher affiliation extraction task, we look for affiliation information tuples got from the Researcher Affiliation Corpus presented in Section 3.4. In the use case presented here, the input is a set of names of researchers who work in a particular research field and the output is a list of affiliations for each researcher. Here, the affiliation is a tuple of *affiliation*, *position* type and start/end *dates* and we treated these attributes as named entities.

6.4.1 Detecting Possible Affiliation Slots

We developed a NER tool for detecting possible actors of a `position` tuple. Note that this task is not a classical NER problem because our goal here is to recognise just those entities which may play a role in a position event. For example, there were a lot of year tokens in the text – having the same orthographic properties – but only a few were related to affiliation information. The contexts of the tokens should play an important role in identifying very narrow semantic NE classes. Table 6.2 lists the frequency of named entities in the 151 paragraphs. As Table 6.2 shows, the vast majority of *affiliation* items are multiword named entities, while only a few *date* attributes consist of more than one word.

	#	Multiword NE	%
affiliation	374	350	93.58
position type	326	159	48.77
year	212	9	4.24

Table 6.2: Frequency of named entities.

To train and evaluate the NER systems, we used each of the 151 paragraphs containing at least one manually labelled position along with 200 other manually selected paragraphs that did not contain any labelled position. We decided to use just these 151+200 paragraphs instead of the full set of 86,735 paragraphs for CPU time reasons. Manual selection – instead of random sampling – was required as there were several paragraphs which contained affiliation information unrelated to the researcher in question, thus introducing noise. In our multi-stage architecture, the NER model trained on this reduced document set was then predicted for the full set of paragraphs and false positives (note that the paragraphs outside the NER training set do not contain any gold standard annotation) had to be eliminated.

We employed the MALLET implementations (McCallum, 2002) of Conditional Random Fields (Lafferty et al., 2001a) presented in Section 4.5, with a general named entity feature set (Szarvas et al., 2006b) for our NER experiments that was described in Section 5.3.2. For the *location*, *organization* and *name* markups given by the NER tool (Szarvas et al., 2006b) trained on the CoNLL-2003 training data set, it achieved F-scores of 89.94 on names, 87.06 on locations and 76.37 on organisations evaluated on the CoNLL-2003 evaluation set (Sang and Meulder, 2003).

Furthermore, the basic named entity feature set was extended using two domain specific gazetteers, namely a list of university names and position types. We should add that a

domain-specific exception list (containing e.g. *Dr*, *PhD*) for improving a general sentence splitter was employed here as well.

Table 6.3 lists the phrase-level $F_{\beta=1}$ results obtained using CRF in the one-researcher-leave-out evaluation scheme, while Table 6.4 lists the results of a baseline method that labels each member of the university and position type gazetteers and identifies years using regular expressions. This comparison highlights the fact that labelling each occurrence of these easily recognisable classes cannot be applied. It gave an extremely low precision score, so contextual information had to be leveraged.

	Precision	**Recall**	**F-score**
affiliation	66.78	53.28	59.27
position type	87.50	70.22	77.91
year	86.42	69.31	76.92
TOTAL	78.73	62.88	69.92

Table 6.3: The results achieved by applying CRF on the Researcher Affiliation Corpus.

	Precision	**Recall**	**F-score**
affiliation	21.43	9.68	13.33
position type	23.27	66.77	34.51
year	65.77	98.99	79.03
TOTAL	32.16	44.08	37.19

Table 6.4: Results of applying the rule-based baseline method on the Researcher Affiliation Corpus.

6.4.2 The Assignment of Subject

When we apply the NER module to unknown documents, we have to decide whether the recognised entities have any connection with the particular person as downloaded pages often contain information about other researchers (supervisors, students, etc.) as well. The subject of the information is generally expressed by a proper noun at the beginning of the page or paragraph and then anaphoric references are used. We assumed here that each position tuple in a paragraph was related to exactly one person and when the subject of the first sentence of the paragraph was a personal pronoun *I*, *she*, *he*, the paragraph belonged to the author of the page.

To automatically find the subject of the paragraphs, we tried out two procedures and evaluated them on the predictions of the NER model introduced earlier. First, we applied an NER model trained on the person names of the CoNLL-2003 corpus (Sang and Meulder, 2003). The names predicted by this method were then compared to the owner of the home-page using name normalisation techniques. If no name was found by the tagger, we assumed that the paragraph belonged to the author. The errors had two sources: the NER trained on an out-domain corpus gave a lot of false negatives and the normalisation method had to deal with incorrect "names" (like *Paul Hunter Curator* as a name phrase) as well.

The second method was simpler. We kept the `position` tuples whose paragraph contained any part of the researcher name or any of the *I*, *she*, *he* personal pronouns. The errors came, for instance, from finding the *Paul* string for *Paul Robertson* in the text snippet *Paul Berger*.

We applied these two subject detection methods to the predictions part of our slot detection NER module. Table 6.5 summarises the accuracy scores of the systems, i.e. whether they made the correct decision on the question "*does this predicted affiliation correspond to the researcher in question?*". The columns of this table show how many affiliation predictions were carried out by the slot detection system, i.e. how often it had to make a decision. We investigated the methods' performance on the paragraphs which contained manually labeled information, on the paragraphs which did not contain any but the slot detection module forecast at least one affiliation here and on the union of these sets of paragraphs. The statistical data listed in the table tell us that the personal pronoun detection approach performs better on paragraphs which actually contain affiliation information. This is due to the fact that this method deletes fewer predictions compared to the name-based one and there are just a few forecasts that have to be removed from the paragraphs which contain useful information.

	#pred	**Name Detection**	**Personal Pronouns**
annotated	165	66.9	87.8
non-annotated	214	71.5	61.2
full set	379	69.4	73.4

Table 6.5: Accuracies of subject detection methods.

To find relationships among the other types of predicted entities (*affiliation*, *position type*, *start year*, *end year*), we applied a simple heuristic. As the `affiliation` slot is the head of the tuple, we simply assigned all the detected entities to the nearest `affiliation` and treated the earlier predicted `year` token as the `start year`. This method made the correct decision in 91.3% and 71.8% of the cases applied on the gold standard annotation and the predicted entities, respectively. We should add that using the predicted labels during the evaluation, the false positives of the NER automatically count as an error in relation detection as well.

6.5 Person Attribute Extraction from Webpages

Besides the problem of researcher affiliation extraction, the extraction of different bibliographical attributes from people's webpages such as the *date of birth*, *affiliation* or *occupation* is also an important web-mining-based NER task as person names are among the most frequently searched items in web search engines (Artiles et al., 2007). However, these types of search results ignore the fact that a name may be associated with more than one person. Sometimes person names are highly ambiguous (see Figure 6.2, which shows three different people with the same name).

Here, we present a web mining system that extracts bibliographical information about persons. The input of this system is the result of web search engine queries in English. Our approach is primarily based on biographical attribute extraction and it uses this information to determine the clusters of persons. Moreover, we were able to evaluate our NER methods

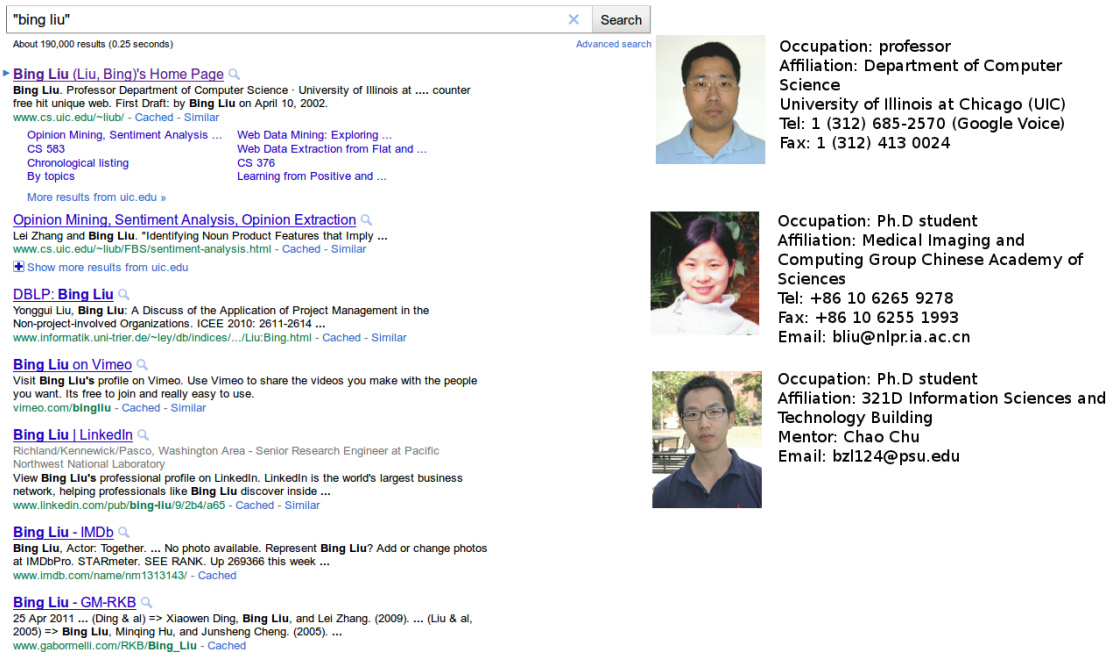

Figure 6.2: The Personal Name Disambiguation Problem.

to mine person-related attributes for persons from webpages, as the English Name Disambiguation Test Corpus (see Section 3.2) contains annotation data about bibliographical items only at the person (cluster) level.

Our system participated in the third WePS challenge (Artiles et al., 2010) and achieved top results on the person attribute extraction subtask.

6.5.1 Named Entity Recognition-based Attribute Extraction

As we mentioned in Section 6.3, usually more pages express content in textual form than in structured form. This is why we concentrated on the natural language-written parts of websites and tables, and we discarded a lot of noisy and misleading elements from pages. In order to identify textual paragraphs, we applied the paragraph extraction rule presented in Section 6.3.1, so we assumed that one piece of text was a textual paragraph if it was longer than 60 characters and it contained more than one verb. In addition, an attribute-specific, relevant section selection method, which was presented in Section 6.3.2, was also applied to filter out the irrelevant paragraphs. Afterwards, we extracted several attributes from relevant sections of webpages using different approaches, like our own NER tool, regular expressions or dictionaries. Lastly, when handling the person disambiguation problem, based on the extracted candidate attributes, we clustered the pages by merging clusters having common person attributes and aggregated attributes with the persons identified.

The task of recognising attributes involved extracting 15 kinds of bibliographical attribute values for target individuals whose names appeared on each of the webpages provided. Moreover, we assumed that different types of attributes can require different types

of extraction approaches. For instance, we were able to extract some well-defined attributes like *phone number* or *e-mail address* using regular expressions or dictionaries. But the automatic identification of more elaborate attributes like *other name* or *place of birth* required more sophisticated NLP solutions. The attribute types and the extraction methods are listed in Table 6.6.

Attribute Class	Examples of Attribute Value	Method
Date of birth	4 February 1888	regexp
Birth place	Brooklyn, Massachusetts	NER
Other name	JFK	NER
Occupation	Politician	NER
Affiliation	University of California, Los Angeles	NER
Award	Pulitzer Prize	NER
School	Stanford University	NER
Degree	Ph.D.	list
Mentor	Tony Visconti	NER
Nationality	American	list
Relatives	Jacqueline Bouvier	NER
Phone	+1 (111) 111-1111	regexp
Fax	(111) 111-1111	regexp
e-mail	xxx@yyy.com	regexp
Web site	http://rgai.inf.u-szeged.hu/	regexp

Table 6.6: Definition of attributes of Person for the WePS attribute extraction task.

In order to automatically extract elaborated attributes from paragraphs, a machine learning-based approach was also used. As we treated these attributes as named entities, the NER tool was applied, which was described in Section 6.4.1. When we investigated the manually annotated attributes in the training and test sets of the second WePS Campaign attribute extraction task (Artiles et al., 2010), we also found that the majority of attributes consisted of multiword named entities. Table 6.7 shows the frequency of multiword named entities in the second WePS training and test sets.

The NER tool was trained on the CoNLL-2003 training data set, where `location`, `person name`, `organisation` and `miscellaneous` named entities were manually annotated. As the majority of bibliographical attributes are a subclass of these classes, we were able to apply the model trained on CoNLL-2003. For instance, the *relatives*, *other name* and *mentor* attributes are subclasses of `person name`. However, some attributes like *occupation* have no manually annotated training dataset to train any NER tool. Hence we automatically generated a training dataset from extracted gold annotations and paragraphs. We used a simple string matching method, but we did not know where the actual attribute occurred in the current text, so the resulting data set was very noisy. The NER tool was trained on this automatically generated data set, and applied on the candidate paragraphs to extract the different attributes.

Attribute Class	#	Multiword NE	%
Affiliation	3,634	2,649	72.89
Award	342	319	93.27
Birthplace	455	344	75.60
Mentor	360	304	84.45
Occupation	3,879	1,369	35.29
Other name	797	669	83.94
Relatives	1,060	757	71.41
School	640	544	85

Table 6.7: Frequency of (multiword) named entities.

6.5.2 Extracting Attribute Classes

Our attribute extraction system consists of two main steps, namely a candidate attribute extraction module and an attribute verification module. With this approach we first mark potential attribute values in a paragraph, then we figure out which candidate values have been found.

When we handle the attribute extraction subtask, it seems necessary to clusterize the attribute classes. Hence, we aggregated similar attributes into logical groups. For instance, the name group contains the *other name*, *relatives* and *mentor* attribute classes. One advantage of this typology scheme is that we can apply the same approach for a logical group. Another advantage is that we can assume priority order among the coherent attributes. For example, we only marked a candidate name as *mentor* if it was not *relatives* or *other name*.

Name	Contact	Organization
relatives	webpage	school
other name	phone number	award
mentor	fax	affiliation

Table 6.8: Attribute typologies

Next, we will elaborate on the extraction procedure for each of the attributes.

Date of birth: If a paragraph contains the phrases *born*, *birth* or *birthday*, we find candidate dates with a date validator within a window of the word. This validator works with 9 different regular expression rules, and can identify dates written in different formats in the span of text.

Birth place: When a candidate paragraph contains *born*, *birth*, *birthplace*, *hometown* and *native* phrases, we use the location markups given by the NER tool. We accept a location as a birthplace if a birthplace validator validates it.

Occupation: According to the WePS2 results, it was one of the most difficult, ambiguous and frequent attribute classes, which is due to the abstract nature of this attribute. Hence we avoided using lists. A NER model was applied to recognise *occupation* from candidate occupation paragraphs, which was trained on the automatically generated training dataset

described in Section 6.5.1.

Organisations (*school*, *award*, *affiliation*): The NER tool was also applied here to identify candidate organization mentions only in affiliation-candidate paragraphs. When the NER model marks a candidate organization phrase, we first search for the *school* attribute. Then a potential candidate organization is marked as a school if it appears near some cue phrases such as *graduate*, *degree*, *attend*, *education* and *science*. Next, we defined a school validator that uses the MIVTU (Watanabe et al., 2009) school word frequency list with *School*, *High*, *Academy*, *Christian*, *HS*, *Central* and *Senior*. We extended this list with the phrases *University*, *College*, *Elementary*, *New*, *State*, *Saint*, *Institute*. First letter capitalised sequences, except for some stopwords like *of* and *at* which contain at least one of these words, were marked as a school by a validator. If the school validator did not validate the potential candidate organization, we looked for the award attribute. When candidate sequences appear near cue phrases such as *award*, *win*, *won*, *receive* and *prize*, we simply assume that the expression with *award* is an attribute. We also defined an award validator that validates a first letter capitalised sequence except for some stopwords like *at* or *of*, if it contains at least one element of the *award*, *prize*, *medal*, *order*, *year*, *player* and *best* phrases. When the candidate string is not a valid *school* and *award*, we tag it to the *affiliation* attribute.

Degree: A list of degrees was compiled manually that contained 62 items. When we found one element from these lists in a paragraph, we marked it as a degree attribute. We assumed that the degree attribute might sometimes be located far from the name in a CV-type webpage.

Names (*relatives*, *other name*, *mentor*): To identify name attributes we also used a NER tool trained on the CoNLL-2003 training dataset. A model extracts name phrases as relatives if they appear in the immediate context of the candidate that indicates family relationships like *father*, *son*, *daughter* and so on. Cue phrases were the same as those in the MIVTU (Watanabe et al., 2009) system used in WePS2 and they are also available in Wikipedia. Sometimes we did not mark the potential candidate sequence for *relatives*, but looked for *other name* attributes instead. We hypothesized that a person does not write his or her name using the same number of tokens; at the same time *other name* has to contain at least a part of the original name. For instance when the original name was *Helen Thomas*, we did not accept the candidate string *Helen McCumber*, but we accepted the *Helen M. Thomas* sequence. This hypothesis may not be true for nicknames. If a name was not marked as *relatives* or *other name*, we checked the potential candidates for a *mentor* name. If it appeared near cue phrases such as *study with*, *work with*, *coach*, *train*, *advisor*, *mentor*, *supervisor*, *principal*, *manager* and *promote*, we marked the potential candidate sequence as a mentor attribute.

Nationality: We created a list of nationalities that contained 371 elements. It had multiple entries for certain nationalities. Once we had found one element from this list in a paragraph or table, we assumed it was a potential nationality attribute. Then we selected the most frequent potential nationality attribute of the webpages.

When extracting *contact* attribute classes, we did not just focus on textual paragraphs, but examined the whole text of webpages as these types of attributes may occur in other parts as well.

Phone: When a text contains *tel*, *telephone*, *ph:*, *phone*, *mobile*, *call*, *reached at*, *of-*

fice, *cell* or *contact* words or a part of the original name, we applied the following regular expression:

```
(((?[0-9+(][.()0-9s/-]4,[0-9])((?(s?x|s?ext|s?hart).?)? d1,5)?)
```

We defined a phone number validator that validated the sequence determined by the regular expression.

Fax: We use the same method as that for phone numbers, except for that we look for the phrases *fax*, *telfax* and *telefax*.

E-mail: We assumed that if somebody offers their e-mail address, it is also a link. Therefore, we examined links that contained the `mailto` tag. Moreover, we assumed that every mail address contains the original name or one part of the original name. Hence we defined an e-mail address validator that validates e-mail addresses. We generate all character trigrams from the original name and when an e-mail address contains at least one of them, the validator accepts it. We defined a stopword list as well. This list contains words such as *wiki*, *support* and *webmaster*. Should a candidate e-mail address contain one from the stopword list, the validator rejected it. Next, we extracted the domain from all accepted e-mail addresses, which we used for the website attribute.

Website: We assumed that when somebody displays a web address on a website, it is also a link, so a web address is a link at the same time. In this case we only extract a website attribute from links. We marked a potential candidate attribute as a website when it contained the original name or one part of the original extracted domain name from the e-mail attribute.

6.5.3 Person Disambiguation

As we mentioned in Section 3.2, only websites that were related to one of two predefined persons were labeled by the annotators in the English Name Disambiguation Test Corpus. Therefore, we had to clusterize the webpages to identify the different websites associated with a person.

Our chief hypothesis in the person disambiguation problem was that it can be effectively solved by using extracted person attributes. Hence, every webpage document was represented by a vector with extracted person attribute values. Then we defined a weighting of attribute classes. In this way, we defined as the most useful attribute classes the *web address*, *e-mail*, *telephone*, *fax number* and *other name* and they got a weight of 3. In addition, we weighted *birth date* as 2, while *birth place*, *mentor*, *affiliation*, *occupation*, *nationality*, *relatives*, *school*, and *award* each got a weight of 1.

In order to determine the similarity between any two different webpages, we need to define a document similarity measure. For this metric, we developed individual normalisation rules for each attribute class. For example, the birth place *United States of America* could be referred to as *USA*, *U.S.A.*, *United States*, *Federal United States*, and so on. To handle this problem, a dictionary of synonyms was created based from on the redirect links of the English Wikipedia, so we could then standardise the different occurrences of well-known named entities. Moreover, some regular expressions or rules were developed to normalise other attribute classes.

Based on the above-described similarity measure, we defined a bottom-up heuristic approach to clustering webpages. As a starting point, each webpage refers to a particular cluster, and then the clusters are merged iteratively until a stopping criterion is reached. For each iteration step, the most similar clusters are merged (the union of their attributes formed the attribute set of the resulting cluster), where the similarity measure of the weighted size of the intersection of the cluster attribute sets was employed. The stopping criterion was defined to be a similarity value threshold of 2, i.e. if the similarity value of the closest clusters is less than 2, the procedure is terminated.

6.5.4 Attribute Extraction Results

As the English Name Disambiguation Test Corpus (see Section 3.2) contains annotation data about bibliographical items, we were able to evaluate our methods to mine person-related attributes for persons got from webpages. In other words, we evaluated the attribute extraction procedure from the clusters of pages belonging to each given person. Our system handles the webpage clustering and person-level attribute extraction tasks together. First, we cluster webpages based on the approach presented in Section 6.5.3 and then select attributes from the clusters related to the associated person.

To train our attribute extraction methods, the WePS2 (Artiles et al., 2009) training and test sets were used, which contained 5,122 websites with 187,032 paragraphs. However, the different attributes have no annotation, so we do not know where the attribute exactly appears in the content. Therefore, we have to map the attributes to the texts of the paragraphs, thereby, the resulting training dataset is noisy. We found 2,781 affiliations, 3,419 occupations and 2,092 biographical paragraphs.

The official evaluation metric applied the attribute recall-based clustering with lenient evaluation. In the case of lenient evaluation, we count as correct all attribute–value pairs judged as correct or inexact by the majority of annotators, and as incorrect otherwise. These results are listed in Table 6.9, where the Xmeans clustering approach was applied, and we defined the minimum number of clusters using heuristic clustering described in Section 6.5.3.

6.6 Extraction of Company Contact Information from Webpages

The third web-mining-based NER problem is the automatic detection of company contact information. Here, we demonstrate a web mining system that can automatically mark the *names* and *addresses* of Hungarian companies on their webpages.

6.6.1 Rule-based Method to Detect Company Contact Information

For the automatic identification of company names and addresses, a rule-based approach was applied. Here we again focus on the textual content of the webpages, hence the proper treatment of misspelling, abbreviations and words with missing accents was necessary when

Attribute Class	Recall	Precision	F-score
affiliation	14.77	26.19	18.88
award	33.33	20.00	25.00
birthplace	48.48	45.71	47.06
date of birth	62.96	50.00	55.74
degree	25.47	38.57	30.68
email	66.67	45.00	53.73
fax	75.00	50.00	60.00
mentor	63.64	18.42	28.57
nationality	43.33	33.33	37.68
occupation	13.26	30.00	18.39
other name	28.99	30.77	29.85
phone	57.58	26.03	35.85
relatives	90.91	16.39	27.78
school	16.67	19.48	17.96
website	40.00	47.06	43.24
all	29.13	32.12	30.55

Table 6.9: Attribute extraction results got on the WePS3 corpus, with lenient annotation and attribute recall based clustering.

we marked the different elements. To solve spelling problems, the Levenshtein-distance was applied, which is able to measure the differences between two sequences. The rule-based method was based on hand-crafted lists and regular expressions, which are listed as follows:

- **name of city:** free tag
  list of cities in Hungary (can contain abbreviations)
- **zip code:** free tag
  regular expression: `[H-]?[1-9][0-9][0-9][0-9]`
- **mail box:** free tag
  regular expression: `(postafiók)?(pf.?)?[1-9]?[0-9]?[0-9]?[0-9]`
- **type of the public place:** semi-free tag (cannot appear before the name of the public place)
  list of the types of Hungarian public places (may contain abbreviations)
- **name of the public place:** semi-free tag (cannot appear before the name of the city)
  list of the names of public places in Hungary (may contain abbreviations)
- **street number:** semi-free tag (cannot appear before the name of public place)
  `[IVXLCDM]*[0-9]*.? emelet [0-9]*`
- **district number:** free tag
  `[IVXLCDM]*[0-9]*.? (ker.?)(kerület.?)`

As a source for the names of public places and a list of Hungarian cities, a database of the Hungarian Post Office was used, which is available on their webpage[1]. In order to achieve better matching results, besides the original phrases, we placed it in the list with abbreviations and without accents. So with the name of the "Dózsa György" public space, the phrases "Dózsa György", "Dozsa Gyorgy", "Dózsa Gy", "Dozsa Gy" also occur in the list of public places.

In the case of the addresses, our rule-based method was based on the fact that addresses must contain some type of public places. Therefore, if the content items of a webpage contain a type of public place, an address in the environment of the detected type of public place was used to find it using the regular expressions and lists mentioned above. In the case of free tags, the different element before and after the type of public space was used to detect it, while in the case of semi-free tags we searched for elements only under certain conditions. Not just whole addresses were marked with this method, but also other items, since there are available incomplete addresses, typically without the ZIP code. Furthermore, if the rule-based method only managed to detect some elements of the address, they were also marked.

The rule-based method for the detection of company names was similar to that for the addresses. In this case our method looked for the types of companies like "kft" or "bt" and the token sequence with capitalised first letters, which is before the identified company type tag, was marked as company name. Table 6.10 shows the results got on the rule-based method on the Hungarian Company Contact Information Web Corpus.

6.6.2 Machine Learning-based Method to Detect Company Contact Information

In order to automatically detect the names and the addresses of companies, a machine learning based approach was also applied. Here, these attributes were also treated as multiword named entities as in Sections 6.4 and 6.5. Therefore, a NER tool was applied, described in Section 6.4.1 with the same basic named entity feature set (Szarvas et al., 2006b). However, we also extended this feature set with problem specific features. The dictionaries were extended with the lists of Hungarian cities, the list of the names of Hungarian public places and the list of the types of Hungarian public places, which were used in the rule-based method. Moreover, the shallow linguistic features were also extended with the regular expressions of the *district number*, *street number*, *mail box* and *zip-code*. So, when the token-sequence in the text matched one pattern typical of *zip code* or *mail box*, the sequence tags were marked as *true*, otherwise they were marked as *false*.

The MALLET implementations of the first-order linear chain CRF classifier (described in Section 5.3.2) were utilized for training our model. The model was evaluated on the Hungarian Company Contact Information Corpus (see Section 3.3) in a 10-fold cross-validation setting. We trained the CRF models with the default settings in Mallet for 200 iterations or until convergence was attained. Table 6.11 shows the results of using this method on the Hungarian Company Contact Information Web Corpus.

[1] goo.gl/C4YC8R

6.6.3 Results on Hungarian Company Contact Information Web Corpus

Tables 6.10 and 6.11 list the results got from using different approaches on the Hungarian Company Contact Information Web Corpus. As the tables show, the name of the company proved to be the most difficult attribute to detect. Here, the machine learning-based method performed better, which is mainly due to the higher recall scores. Overall, we found that the machine learning-based approach achieved better F-scores – except the ZIP code – than those for the rule-based method. Although the difference was not essential, the performance of the CRF-based method mainly was attributed to the better precision scores, except for the name of the company, when the rule-based method yielded a higher precision score.

Attribute Class	Recall	Precision	F-score
street number	71.00	80.15	74.12
ZIP code	86.88	84.50	85.67
name of street	77.69	81.87	79.62
name of city	80.63	83.24	81.91
name of company	22.35	43.77	29.59

Table 6.10: Results obtained for the rule-based method for attributes in terms of recall, precision and F-score on the Hungarian Company Contact Information Web Corpus.

Attribute Class	Recall	Precision	F-score
street number	71.69	81.35	76.21
ZIP code	80.75	86.53	83.54
name of street	77.38	88.60	82.61
name of city	77.65	89.09	82.98
name of company	29.58	37.33	33.01

Table 6.11: Results obtained for the machine learning-based method for attributes in terms of recall, precision and F-score on the Hungarian Company Contact Information Web Corpus.

6.7 Discussion

In the case of the researcher affiliation extraction task, the machine learning-based system achieved remarkably better results than those got by the baseline method. We showed experimentally that it could exploit the contextual information and that the labelled entities were those that were affiliation-related. Also, our person-related information extraction method was able to efficiently extract the different types of attributes from webpages, and we got the best results on the WePS3 challenge.

We manually analysed the errors on a part of the Researcher Affiliation Corpus and found that some typical errors were present. The annotation guide said that the geographical location of the affiliation was a part of the affiliation as it sometimes identifies the department

(e.g. *Hewlett-Packard Labs in Palo Alto*). This extension of the phrase proved to be problematic because there were several cases with the same orthographic features (e.g. *Ph.D. from MIT in Physics*). The acronyms immediately after the affiliation are a similar case, which we regard as part of the name and the NER cannot easily handle it (e.g. *Centre for Policy Modelling (CPM)*). As there is no partial credit, an incorrect entity boundary is penalised both as a false positive and as a false negative.

These points also explain the surprisingly low precision scores of the baseline system we got as it labelled university names without more detailed identification of the unit (e.g. *Department of Computer Science,* $[Waterloo\ University]_{BASELINE}$). We should add that these two annotation principles are questionable, but we expect that information might get lost without them. Moreover, there is another reason for the low recall, which is that our human annotators found textual clues for position types on verbs as well (e.g. *I* $lead_{TYPE}$ *the Distributed Systems Group*). The context of these labelled examples is clearly different from that of the usual position type.

Comparing the two subject detection methods, we see that the name detection model which learnt on an out-domain corpus made a lot of mistakes, thus the method based on it judged more paragraphs as irrelevant ones. The name detection could be improved by a domain corpus (for instance the training corpus did not contain any Prof. NAME case) and by applying more sophisticated name normalisation techniques. When we manually analysed the errors of these procedures we found that each false negative of the simpler subject detection method was due to the errors of the textual paragraph identification method used. There were several itemisations whose header was the type *"Previously I worked for:"* and the textual items themselves did not contain the subject of the affiliation information. The false positives often originated from pages which did not belong to the researcher in question, but contained his name (e.g. *I am a Ph.D. Student working under the supervision of Prof. NAME*).

Next, an error analysis of the affiliation head seeking heuristic revealed that 44% of the predicted `position type` and `year` entities' sentences did not contain any `affiliation` prediction. With the gold standard labelling, there were 6 sentences without `affiliation` labels and only one of them used an anaphoric reference, and the others were consequences of the erroneous automatic sentence splitting of the HTML documents. The prediction of the NER system contained many more sentences without any `affiliation` label. These could be fixed by forcing a second forecast phase to predict `affiliation` in these sentences or by removing these labels in a post-processing step.

As we saw in the researcher affiliation extraction problem, the detection of the correct boundaries of NEs is not a trivial task. Moreover, the classification of the entities is also a hard task in machine learning-based approaches, since the person-related information extraction task has 15 different attributes. Furthermore, some attributes are included in one semantic class, like *mentor*, *other name*, or *relatives* are person names. We were able to extract this type of attributes in a better way via machine learning approaches when we placed the attribute classes into different logical groups and we were able to assume ordered relations among the coherent attributes. However, some attributes like contact attributes do not require complex solutions to detect them in webpages. Yet, it is slightly surprising that the

recall of this type of attributes is not close to 100%. We identified several reasons for this. One is that some *e-mail addresses* are created by Java scripts and it is not easy to detect them in the source page. Furthermore, if a page contains a list of a large number of names with availability information like *e-mail address* and *phone number* in a table or other format, it is not trivial to find the proper e-mail of the target person. The other problem is that there are many e-mail addresses mentioned in webpages, including e-mail addresses of the web masters, contact persons, friends, persons who make comments, so we need a smart way to filter out these e-mail addresses.

The identification of *affiliation*, *award* or *occupation* also proved to be a difficult task as many people have a wide variety of attribute names. For example, a soccer player's affiliation, such as *Barcelona F.C.* or *Hungarian national football team* is quite difficult to detect, while a university professor affiliation is for example *University of Szeged*.

As Tables 6.10 and 6.11 show, the automatic detection of the *address* attribute is a much easier task than extracting *name of companies*. Basically, the attributes of the addresses are well-defined, while the names of companies are diverse. The names can contain non-Hungarian words *(Shop Builder bt.)*, acronyms *(MOL Nyrt.)*, abbreviations (*KNB FUENTE Ingatlanforgalmazó és Tanácsadó Kft.*) or company names with English abbreviations (*Stancforma Ltd.*), while the tokens can contain digits (*Flow2000 Bt.*) and non-alphabetic characters. As the rule-based method only focused on sequences with a capitalised first letter, it could not detect the names of companies which contain conjunctions like *Majer és Majer kft.* Owing to this fact, the machine learning-based method can achieve a higher recall score.

It is interesting that the F-score got from the Person Disambiguation problem on the WePS3 corpus was considerably worse than those got on the other two web-mining NER problems. However, we solved a harder task here, as we did not just detect the different named entities in webpages, but we had to assign which attributes were related to the current person and which attribute was related to someone else. Moreover, the machine learning-based models that we applied outperformed our rule-based baseline methods, which underlines the fact that our machine learning-based approaches can be suitably applied to NER from web contents.

6.8 Summary of Thesis Results

In this chapter, we presented our methods for recognising different types of named entities from webpages. The main findings of this chapter are the following:

- Here, we presented three different Named Entity Recognition problems from the area of web mining, in two different languages, namely English and Hungarian.
- As we found that most of the useful information was available in natural text format in webpages, **we focused on the raw textual parts of the webpages** instead of the structured parts.

- As only a small portion of extracted textual paragraphs contained useful information, we **developed attribute-specific relevant section selection modules.** Our filtering method cleverly exploited the paragraphs containing a current attribute (`positive paragraphs`).

- We treated **named entities similarly to nominal compounds as they form one semantic unit**, consist of more than one word and function as a noun. Also, we found in three NER datasets that the majority part of named entities were multiword named entities. Therefore similar machine learning-based methods could be applied just as we did in the case of nominal compounds in Chapter 5.

- We were able to extract attributes belonging to the same semantic class in a better way via machine learning approaches when **we placed the attribute classes into logical groups** and we assumed ordered relations among the coherent attributes.

- We presented a Web Content Mining system for **gathering affiliation information** from the homepages of researchers.

- Our attribute extraction method efficiently **extracted the different types of person-related attributes** from webpages and we achieved top results in the WePS3 challenge.

- We presented our approaches to detect **names and addresses of companies with rule-based and machine learning-based methods**.

In Nagy et al. (2009), information about researchers' affiliations is identified from webpages. The remaining parts of this chapter are solely the author's work. The problem of person attribute extraction from webpages is described in Nagy T. (2012). The author participated in the third WePS challenge (Artiles et al., 2010) and achieved top results on the person attribute extraction subtask. The extraction of company contact information is presented in Nagy T. (2009).

Chapter 7

Sequence Labeling for Detecting English and Hungarian Light Verb Constructions

In the previous chapters we presented our different sequence labeling-based methods for the automatic detection of multiword NEs and NCs. Here, we present our conditional random fields-based tool for identifying verbal light verb constructions in running texts. To demonstrate the flexibility of our tool, we experimented on two, typologically different languages, namely English and Hungarian.

Furthermore, different types of texts may contain different types of light verb constructions, and the frequency of light verb constructions may differ from domain to domain. Hence we will focus on the portability of models trained on different corpora and we also investigate the effect of simple domain adaptation techniques to attempt to reduce the gap between the domains. Our results show that in spite of their special domain characteristics, out-domain data can also contribute to successful LVC detection in different domains.

7.1 Related Work

Now, we will present related work on detecting light verb constructions in running texts.

7.1.1 Approaches to Identifying Light Verb Constructions

There are two basic approaches for identifying LVCs. In the first approach, we attempt to classify LVC candidates, which means that we extract LVC candidates (usually verb-object pairs including one verb from a well-defined set of 3-10 verbs) from texts and then they make a binary decision whether they are LVCs or not (Stevenson et al., 2004; Tan et al., 2006; Fazly and Stevenson, 2007; Van de Cruys and Moirón, 2007; Gurrutxaga and Alegria, 2011). In the second approach, other studies identified LVCs in running texts, having taken contextual information into account (Diab and Bhutada, 2009; Tu and Roth, 2011; Vincze et al., 2011a; Nagy T. et al., 2011b). While the first approach assumes that a specific candidate is an LVC or not independently of context, the second one may take account of the fact that there are contexts where a given candidate functions as an LVC whereas in other

contexts it does not, due to structural or morphological homonymies.[1] Compare *the government will **make decisions** on foreign policy issues* vs. *they will **make decisions** taken by the government publicly available* or ***számba vettem** a lehetőségeket* (consideration-ILL take-PAST-1SGOBJ the possibility-PL-ACC) "I considered the possibilities" vs. ***számba vettem** a nyalókát* (mouth-1SGPOSS-ILL take-PAST-1SGOBJ the lollipop-ACC) *"I put the lollipop into my mouth"*, where the first occurrences of *make decisions* and *számba vettem* are LVCs and the second ones are not. Hence, the output of the first approach is a list of LVCs extracted from the text, while the second method produces raw texts where LVCs are automatically labeled.

With our annotated corpora at hand (see Section 7.2.1), we were able to examine the proportion of LVC and non-LVC uses of some specific LVC candidates. For instance, the phrase *tárgyalást folytat* (negotiation-ACC continues) usually means "to conduct a negotiation", which is an LVC, but in certain contexts it can mean "to continue a(n ongoing) negotiation", which is not an LVC. In the corpora, there are 13 LVC uses and 1 non-LVC use. However, the sequence *megbeszélést tart* (meeting-ACC holds) "to have a meeting" – which can be also treated as an LVC (out of context) – occurs only once in the corpus, and in a non-LVC use: *megbeszélést tart célszerűnek* (meeting-ACC holds necessary-DAT) "he thinks that a meeting is required". Thus, while non-LVC usage of LVC-candidates is not so frequent, the corpora do indeed contain some examples.

In this chapter, we identify LVCs in running texts, that is, we follow the second approach and carry out a token-based identification of LVCs instead of a type-based one. In other words, we decide whether the given sequence of words is an LVC or not in a certain context.

7.1.2 Methods for Identifying Light Verb Constructions

Several applications have been developed for identifying MWEs and LVCs, which can be classified according to the methods they apply (Piao et al., 2003; Dias, 2003). First, statistical models rely on word frequencies, co-occurrence data and contextual information to decide whether a bigram or trigram (or even an n-gram, i.e. a sequence of words) should be treated as a multiword expression or not. For more details see e.g. Bouma (2010), Villavicencio et al. (2007). Statistical systems can be easily adapted to other languages and other types of multiword expressions, but they are not able to identify rare multiword expressions, which is the main drawback of these methods as about 70% of multiword expressions occur only once or twice in a large corpus (Piao et al., 2003; Vincze, 2011). As for LVC detection, Stevenson et al. (2004), Fazly and Stevenson (2007), Van de Cruys and Moirón (2007) and Gurrutxaga and Alegria (2011) built their system on statistical features, among others. Stevenson et al. (2004) focused on deciding whether true LVC candidates containing the verbs *make*, *take* or *give* are acceptable or not. Fazly and Stevenson (2007) used linguistically motivated statistical measures to distinguish subtypes of verb + noun combinations. Van de Cruys and Moirón (2007) described a semantic-based method for identifying verb-preposition-noun combinations in Dutch. Their method relies on selectional preferences for both the noun and

[1] An intermediate solution is that of `mwetoolkit` (Ramisch et al., 2010b; Ramisch et al., 2010a), which provides a list of MWEs extracted from texts. Hence, MWE candidates that occur at least once as an MWE within the text are treated as MWEs. However, here non-MWE uses of the same unit are ignored.

the verb and they also utilize automatic noun clustering when considering the selection of semantic classes of nouns for each verb. Gurrutxaga and Alegria (2011) extracted idiomatic and light verb noun + verb combinations from Basque texts by employing statistical methods. Since Basque is a free word-order language, they hypothesized that a wider window would yield more significant cooccurrence statistics, but their initial experiments did not confirm this.

Other studies employ rule-based systems in LVC detection (Diab and Bhutada, 2009; Nagy T. et al., 2011b; Vincze et al., 2011a; Sinha, 2011), which are usually constructed on the basis of (shallow) linguistic information. Diab and Bhutada (2009) used a supervised system for classifying verb-noun combinations as literal or idiomatic in context. Vincze et al. (2011a) exploited shallow morphological features for identifying LVCs in English texts, while the domain specificity of the problem was highlighted in Nagy T. et al. (2011b). Sinha (2011) found that linguistic-based information can help when identifying Hindi multiword expressions in an English–Hindi parallel corpus.

Some hybrid systems make use of both statistical and linguistic information as well (Dias, 2003; Tan et al., 2006; Bannard, 2007; Cook et al., 2007; Tu and Roth, 2011; Samardžić and Merlo, 2010), which results in better recall scores. Dias (2003) presented a system which was based on word statistics and information got from POS-tagging and syntactic parsing. Tan et al. (2006) tried to identify true LVCs by applying machine learning techniques. They found that in this task it is especially the random forest classifier that could efficiently combine statistical and linguistic features. Bannard (2007) sought to identify verb and noun constructions in English on the basis of syntactic fixedness. He examined whether the noun could have a determiner or not, whether the noun could be modified and whether the construction could have a passive form, which features were exploited in the identification of the constructions. Cook et al. (2007) differentiated between literal and idiomatic uses of verb and noun constructions in English. Their basic hypothesis was that the canonical form of each construction occurs mostly in idioms as they show syntactic variation to a lesser degree than constructions in literal usage. Samardžić and Merlo (2010) analyzed English and German LVCs in parallel corpora: they paid special attention to their manual and automatic alignment. They found that linguistic features (i.e. the degree of compositionality) and the frequency of the construction both have an impact on the alignment of the constructions. Tu and Roth (2011) classified verb + noun object pairs as being LVCs or not, using a Support Vector Machine. They employed both contextual and statistical features and concluded that on ambiguous examples, local contextual features perform better.

Sass (2010) developed a method for extracting multiword verbs from parallel corpora. By aligning the verbs in parallel clauses, a complex verb is produced to which arguments are ordered with tags denoting the language of the subcorpus it came from. From these representations the original algorithm is able to detect the multiword verbs for each language of the parallel corpus, along with cases where a multiword verb corresponding to a single word verb in the other language can also be extracted.

Rule-based or hybrid methods may be highly language-dependent because of the amount of linguistic rules encoded in them, so it is costly to adapt them to different languages or even to different types of multiword expressions. Still, the combination of different methods

may improve the performance of systems for LVC detection (Pecina, 2010).

As for Hungarian, we are aware of one system that identifies multiword verbs (LVCs and idioms) (Sass, 2013); however, it does not distinguish between the two classes. Here, we argue that it is important to separate LVCs and idioms because LVCs are semi-productive and semi-compositional – which may be exploited in applications like machine translation or information extraction (Vincze, 2011) – in contrast to idioms, which have neither feature.

7.2 Experiments

Now we will present our methodology and our results on detecting verbal LVCs.

7.2.1 Domain Specificities of Light Verb Constructions in Corpora

In our experiments, three corpora for both English and Hungarian were used. When choosing the corpora we kept in mind the fact that the same domains would be employed for both languages, so interlingual comparisons across domains could be made as well. Thus, we selected texts from the newspaper, short news and law domains.

For the English newspaper domain, we selected the English versions of texts from bilingual magazines from SzegedParalellFX (see Section 3.7). However, JRC-Acquis (see Section 3.10) was selected for the English law domain and the CoNLL-2003 dataset (see Section 3.11) for the English short news domain. As for Hungarian, from the subcorpora of Szeged TreebankFX (see Section 3.6), the domains of law, short business news and newspaper texts were chosen.

To compare the performance of our system with others, we evaluated our method on the Tu&Roth dataset (presented in Section 3.8) as well.

In order to confirm the domain specificity of detecting LVCs, we carried out a detailed data analysis on the LVCs occurring in the corpora. First, LVCs were gathered from the corpora and lemmatized and the frequency of each lemma was calculated. Data values are presented in Table 7.1, and Tables 7.2 and 7.3 list the most frequent LVCs in each corpus. As can be seen, the distribution of LVCs in the corpora varies somewhat: the top 10 LVCs are responsible for only 17.6% and 25.7% of the LVC occurrences in the CoNLL-2003 and SzegedParalellFX corpora, respectively, while this value is 50% in the JRC-Acquis corpus. As for the Hungarian case, the situation is similar: the 10 most frequent LVCs represent 49.5% of the LVCs in the law subcorpus, whereas it is only 31.4% and 23.4% in the short news and newspaper subcorpora, respectively.

We also investigated the extent to which the corpora overlap, i.e. how many LVCs occur in each corpus or in at least two of the corpora. The Dice and Jaccard distances between the corpora were also calculated on the basis of the union and intersection of the LVCs found in the corpora. Table 7.4 lists these values. We only found 11 LVCs that occur in each of the English corpora and 28 that occur in each of the Hungarian corpora, which aptly underlines the domain specificity of the problem; namely, different corpora contain different LVCs. This fact enables us to apply domain adaptation techniques.

Corpus	Verbal LVCs	Lemmas	Occurrences of lemmas
SzegedParalellFX	354	216	1.64
JRC-Acquis	204	85	2.40
CoNLL-2003	235	173	1.36
SzT newspaper	453	238	1.90
SzT law	629	167	3.77
SzT short news	563	236	2.29

Table 7.1: Statistical data on LVCs in the corpora.

	SZPFX-newspaper		JRC-Acquis		CoNLL-2003	
1.	take place	25	enter into force	27	take place	7
2.	play a role	17	take into account	18	give detail	6
3.	give a concert	9	take account	12	play a game	6
4.	take a look	7	meet the requirements	11	catch fire	4
5.	take part	7	take place	9	fall short	3
6.	spend time	6	take measure	7	have an impact	3
7.	have an effect	5	carry out an activity	5	make a debut	3
8.	make a debut	5	play a role	5	play cricket	3
9.	pay attention	5	deliver an opinion	4	take a step	3
10.	take care	5	give a judgment	4	take part	3

Table 7.2: The most frequent English LVCs.

	SzT newspaper		SzT law		SzT short news	
1.	részt vesz “to take part”	31	sor kerül “the time has come”	109	nyilvánosságra hoz “to publish”	40
2.	sor kerül “the time has come”	14	lehetőséget ad “to offer a possibility”	37	hírül ad “to make a report”	38
3.	őrizetbe vesz “to take into custody”	11	szerződést köt “to make a contract”	31	ajánlatot tesz “to make an offer”	28
4.	szerződést köt “to make a contract”	10	sor kerül “the time has come”	29	tárgyalást folytat “to conduct a negotiation”	18
5.	szert tesz “to get access”	8	eleget tesz “to fulfill”	23	szerződést köt “to make a contract”	13
6.	lehetőséget ad “to offer a possibility”	7	forgalomba hoz “to put into circulation”	19	megállapodást köt “to make an agreement”	11
7.	támogatást kap “to receive support”	7	határozatot hoz “to make a verdict”	17	megbízást ad “to give an assignment”	8
8.	döntést hoz “to take a decision”	6	nyilvánosságra hoz “to publish”	17	döntést hoz “to take a decision”	7
9.	helyet kap “to get space”	6	igényt tart “to have a claim”	15	eleget tesz “to fulfill”	7
10.	igénybe vesz “to take up”	6	részt vesz “to take part”	15	feljelentést tesz “to make an accusation”	7

Table 7.3: The most frequent Hungarian LVCs.

Corpora	Intersection	Dice	Jaccard
JRC-CoNLL	18	0.1395	0.9250
JRC-Paralell	17	0.1130	0.9400
Paralell-CoNLL	27	0.1388	0.9254
SzT law-SzT news	41	0.2035	0.8867
SzT law-SzT paper	52	0.2568	0.8180
SzT paper-SzT news	73	0.3080	0.8527

Table 7.4: Distance between the corpora.

	Wiki50			**SzegedParalellFX**			**Tu&Roth**
Approach	**Recall**	**Precision**	**F-score**	**Recall**	**Precision**	**F-score**	**Accuracy**
Own method	44.56	74.55	55.78	46.48	72.70	56.71	73.93
Tu&Roth	–	–	–	–	–	–	68.52

Table 7.5: Results of different methods in terms of recall, precision, F-score and accuracy in different corpora. **Own Method:** results of own method. **T&R:** results of Tu and Roth (2011) in terms of accuracy

7.2.2 Sequence Labeling-based Method

As Tables 3.4, 3.5 and 3.6 show, verbal LVCs are the most frequent among the different types of LVCs in both English and Hungarian cases. In order to avoid sparsity problems here we just focus on identifying verbal LVCs in running text and we can consider this task as a sequence labeling problem as verbal LVCs are contiguous.

As we mentioned in Section 4.5, CRF is one of the most effective method to solve sequence labeling tasks. Therefore, we trained the MALLET implementations (McCallum, 2002) of the first-order linear chain Conditional Random Fields (CRF) classifier (Lafferty et al., 2001b) with the feature set detailed in Section 7.2.3 and evaluated it on the English and Hungarian corpora in a 10-fold cross-validation setting, taking document boundaries into account. We also evaluated our method in a leave-one-document-out scheme on the Wiki50 and the whole SzegedParallelFX corpora as English LVCs were also manually annotated. The results of applying this approach are shown in Table 7.5. We trained the CRF models with the default settings in Mallet for 200 iterations or until convergence was reached. As evaluation metrics, we employed $F_{\beta=1}$ scores at phrase level.

To compare the performance of our system with others, we evaluated it – with the necessary modifications (e.g. detecting only true light verb constructions) – on the Tu&Roth dataset (Tu and Roth, 2011) too. We were able to achieve an accuracy score of 73.93%, which is 5.41% higher than that achieved with the Tu&Roth method (Tu and Roth, 2011) (68.52%).

7.2.3 Feature Set

For the automatic identification of LVCs in corpora, we implemented a machine learning approach, which we elaborate upon below. Our tool is based on a general named entity

feature set (Szarvas et al., 2006b), which was presented in Section 5.3.2.

This basic feature set was implemented for Named Entity Recognition. Since LVCs never contain named entities, these features may also contribute to the overall performance; however, we extended this basic feature set with LVC specific features. We classify these LVC specific features according to the following categories: orthographic, lexical, morphological, statistical, syntactic and semantic.

Orthographic features: The **suffix** feature is also based on the fact that many nominal components in LVCs are derived from verbs. This feature checks whether the lemma of the noun ended in a given character bi- or trigram.

Lexical features: We exploit the fact that the **most common verbs** are typically light verbs. Therefore, fifteen typical light verbs were selected from the list of the most frequent verbs taken from Wiki50 in the case of English and from the subcorpora of Szeged TreebankFX that were not used in our experiments in the case of Hungarian. But for the Tu&Roth dataset we just used their six light verbs (*do*, *get*, *give*, *have*, *make* and *take*). Next, we investigated whether the lemmatised verbal component of the candidate was one of these fifteen/six verbs. The **lemma of the noun** was also applied as a lexical feature. The nouns found in LVCs were collected from the above-mentioned corpora. Afterwards, we constructed **lists of lemmatised LVCs** got from the above-mentioned corpora. We used it as a binary feature whether or not the LVC candidate occurred in the lists.

Morphological features: As the nominal component of LVCs is typically derived from a verbal stem (*make a decision*) or coincides with a verb (*have a walk*), the **VerbalStem** binary feature focuses on the stem of the noun; if it had a verbal nature, the candidates were marked as *true*. The **POS-pattern** feature investigates the POS-tag sequence of the potential LVC. If it matched one pattern typical of LVCs (e.g. `verb + noun`) the candidate was marked as *true*; otherwise as *false*. For morphological and dependency parsing we applied the Bohnet parser (Bohnet, 2010) for English and `magyarlanc 2.0` (Zsibrita et al., 2013) for Hungarian. As Hungarian is a morphologically rich language, we selected those morphological features that seemed to play an important role in determining whether an LVC candidate is a genuine LVC in context (**'SubPOS'**) or not and unnecessary features were deleted from the representation. For instance, the number and person features of a verb are irrelevant for LVC detection and were thus neglected. The **'productDeriv'** feature was used to detect non-productive derivations in Hungarian, in the case of those nouns that were derived historically from a verb but the derivational suffix is not considered to be productive any more.

Statistical features: We also extended the English feature list with potential English LVCs and their **occurrences**. We trained a CRF classifier with our LVC specific features on the Wiki50 corpus and extracted potential LVCs from 10,000 Wikipedia pages. We created the **prediction list** from the most frequent LVCs. This list contained 424 different potential LVCs and we examined whether the LVC candidate occurred in the list (binary feature).

Syntactic features: The dependency label between the noun and the verb was added as a feature.

Semantic features: Here, we specified the **other entities in the sentence**, like NEs and nominal compounds, which were also used as features. We employed the Stanford Named

Entity Recognition tool (Finkel et al., 2005) and detected nominal compounds following the methods presented in Section 5.3.2.

7.2.4 Domain Adaptation

As three different domains were available for both languages, we were able to carried out cross-domain and domain adaptation experiments. We used a pure cross-domain (CROSS) setting where our model was trained on the source domain and evaluated on the target (i.e. no labeled target domain datasets were used for training); e.g. we trained the model on Szeged-ParalellFX and tested it on JRC-Acquis. We also examined how domain adaptation could enhance the results if we only have a limited amount of annotated target data at hand. Domain adaptation is especially useful when there is only a limited amount of annotated data available for one domain, but there is plenty of data for another domain. Using the domain with a lot of annotated data as the source domain and a domain with limited data as the target domain, domain adaptation techniques can successfully contribute to the learning of a model for the target domain (see e.g. Daumé III (2007)).

A very simple approach was used for domain adaptation: the training dataset was extended with sentences taken from the target. First, we extended the training dataset with 500 target sentences, then kept adding 500 sentences until we reached 3000. To evaluate the domain adaptation, we performed a 10-fold cross-validation at the document level by training on the union of the source data and the sentences selected from the target domain (DA). For each fold, 10% of target data was used for testing and additional sentences for training were randomly selected from the sentences not used for testing. We also investigated what could be achieved if the system was trained only on the added target sentences without using the source domain in the training process (ID). This model was evaluated in a 10-fold cross-validation setting at the document level. In the pure in-domain setting, we also performed 10-fold cross-validation at the document level on each domain (TARGET) (i.e. only the target domain was used for both of training and testing).

As a baseline, we applied simple dictionary-lookup (DL). Texts were lemmatized and if an item taken from the lists used by the LVC list feature occurred in the text, it was marked as an LVC. We also compared our results with those of a rule-based LVC recognition method (RB) (Vincze et al., 2011a), which relies heavily on POS-rules. It means that each n-gram that matched the pre-defined patterns was accepted as an LVC, just like our POS-pattern feature. As this method provides a big pool of potential LVCs, they were filtered by applying some additional criteria: the same Suffix, Lemma and Syntax features were applied that were presented in Section 7.2.3. The results of our experiments can be seen in Tables 7.6, 7.7, 7.8 and 7.9.

7.3 Results

Tables 7.6 and 7.7 give the results for the English corpora, while Tables 7.8 and 7.9 show the corresponding results for the Hungarian corpora. The domain adaptation results were obtained by extending the source domain with 3000 sentences taken from the target domain.

Corpus	TARGET	RB	DL	Diff$_{RB}$	Diff$_{DL}$
JRC-Acquis	64.09	39.93	25.78	-24.16	-38.31
CoNLL-2003	59.41	47.63	13.79	-11.78	-45.62
SZPFX	62.50	44.06	20.50	-18.44	-42.00
Avg.	62.00	43.87	20.02	-18.13	-41.98

Table 7.6: Experimental results got on English corpora in terms of F-score. **TARGET:** in-domain setting. **RB:** rule-based methods. **DL:** dictionary lookup. **Diff$_{RB}$:** differences between the TARGET and RB results. **Diff$_{DL}$:** differences between the TARGET and DL results.

Table 7.6 shows the results on the English corpora. Based on the 10-fold cross validation results, our system was the most effective in the case of the legal domain JRC-Acquis (F-score = 64.09). At the same time the CoNLL-2003 short news domain proved to be the most difficult corpus, where the F-score was only 59.41. In the case of cross-validation experiments, the best results were got for the JRC-Acquis legal domain. The average results of the CROSS experiments of the three different corpora were F-scores of 10.16 lower than the corresponding TARGET results. The rule-based (RB) approach proved to be the best on CoNLL-2003 with an F-score of 47.63. The difference between the average results of the TARGET and the RB experiments was 18.13. The results got from using the baseline dictionary-lookup method were noticeably exceeded by the TARGET results.

The DA column of Table 7.7 lists the results obtained for the English domain adaptation task. Domain adaptation was the most effective when SzegedParalellFX was the target corpus. The domain adaptation results exceeded the cross-validation experiments by 7.44. The average difference between in-domain and domain adaptation experiments was 5.73.

Table 7.8 shows the baseline and 10-fold cross-validation target results for the Hungarian corpora. Our system proved to be the most effective for the legal domain (with an F-score of 78.97). The average CROSS F-score was 13.46 lower than that for the TARGET scores. The rule-based approach proved to be the best on the SzT law corpus with 58.56. The dictionary lookup achieved 31.6 on the three corpora, which was exceeded by the TARGET results that yielded an F-score of 35.54.

Table 7.9 lists the results for Hungarian domain adaptation. Based on these scores, domain adaptation proved to be the best (better by 15.97%) when SzT law was the source and SzT news was the target. The average domain adaptation results were 8.76 percentage points higher than the CROSS results. The average difference between in-domain and domain adaptation results was 4.55.

The size of the target data added to the source datasets definitely influenced the results, as shown in Figures 7.1 and 7.2 with two typical settings. The first part of the diagram shows the JRC-Acquis target results, cross experiments, baselines and domain adaptation results got when the source was CoNLL-2003. This model can outperform the JRC-Acquis TARGET result when we just add 1500 target sentences to the training data and the F-score was 3.95 better when 3000 target sentences were added. The gap between the in-domain and domain adaptation results progressively decreases with the size of the dataset. But the

Source	Target	TARGET	CROSS	DA	ID	Diff$_{CROSS}$	Diff$_{DA}$	Diff$_{DA/ID}$
SZPFX	JRC-Acquis	64.09	59.05	67.04	59.88	-5.04	7.99	7.16
SZPFX	CoNLL-2003	59.41	47.35	51.61	43.37	-12.06	4.26	8.24
JRC-Acquis	SZPFX	62.50	50.83	61.92	60.99	-11.67	11.09	0.93
JRC-Acquis	CoNLL-2003	59.41	44.38	52.05	43.37	-15.03	7.67	8.68
CoNLL-2003	JRC-Acquis	64.09	57.59	68.04	59.88	-6.50	10.45	8.16
CoNLL-2003	SZPFX	62.50	51.84	62.14	60.99	-10.66	10.30	1.15
Avg.	-	62.00	51.84	60.47	54.74	-10.16	8.63	5.73

Table 7.7: Domain adaptation results on English corpora in terms of F-score. **TARGET:** in-domain setting.**CROSS:** cross-domain setting. **DA:** domain adaptation setting. **ID:** training on a limited set of target data. **Diff**$_{CROSS}$**:** differences between the TARGET and CROSS results. **Diff**$_{DA}$**:** differences between the CROSS and DA results. **Diff**$_{DA/ID}$**:** differences between the DA and ID results.

Source	Target	TARGET	CROSS	RB	DL	Diff$_{CROSS}$	Diff$_{RB}$	Diff$_{DL}$
SzT news	SzT paper	53.51	52.07	39.80	32.72	-1.44	-13.71	-20.79
SzT news	SzT law	78.97	67.85	58.56	33.50	-11.12	-20.41	-45.47
SzT paper	SzT news	68.94	51.93	36.70	28.57	-17.01	-32.24	-40.37
SzT paper	SzT law	78.97	68.74	58.56	33.50	-10.23	-20.41	-45.47
SzT law	SzT news	68.94	43.61	36.70	28.57	-25.33	-32.24	-40.37
SzT law	SzT paper	53.51	37.85	39.80	32.72	-15.66	-13.71	-20.79
Avg.	-	67.14	53.67	45.02	31.60	-13.46	-22.12	-35.54

Table 7.8: Experimental results on different source and target Hungarian domain pairs in terms of F-score. **TARGET:** in-domain setting. **CROSS:** cross-domain setting. **RB:** rule-based methods. **DL:** dictionary-lookup. **Diff**$_{CROSS}$**:** differences between the TARGET and CROSS results. **Diff**$_{RB}$**:** differences between the TARGET and RB results. **Diff**$_{DL}$**:** differences between the TARGET and DL results.

Source	Target	CROSS	DA	ID	Diff$_{DA}$	Diff$_{DA/ID}$
SzT news	SzT paper	52.07	55.08	46.89	3.01	8.19
SzT news	SzT law	67.85	74.00	74.18	6.15	-0.18
SzT paper	SzT news	51.93	62.21	52.57	10.28	9.64
SzT paper	SzT law	68.74	71.96	74.18	3.22	-2.22
SzT law	SzT news	43.61	59.58	52.57	15.97	7.01
SzT law	SzT paper	37.85	51.76	46.89	13.91	4.87
Avg.	-	53.67	59.97	54.20	8.76	4.55

Table 7.9: Domain adaptation results on Hungarian corpora in terms of F-score. **DA:** domain adaptation setting. **ID:** training on a limited set of target data. **Diff**$_{DA}$**:** differences between the CROSS and DA results. **Diff**$_{DA/ID}$**:** differences between the DA and ID results.

Feature	Recall	Precision	F-score	Difference
Dictionary-lookup	20.81	45.56	28.57	-
Base features	42.73	73.25	53.98	-
All features	60.82	79.58	68.94	-
LVC Lists	55.32	76.10	66.46	-2.48
POS-pattern	58.33	79.66	67.35	-1.59
SubPos	57.98	78.42	66.67	-2.27
Syntax	59.04	78.35	67.34	-1.60
Lemma	60.28	77.63	67.86	-1.08
Suffix	57.62	78.50	66.46	-2.48
VerbalStem	60.11	79.85	68.48	-0.46
productDeriv	60.06	80.05	68.62	-0.32
RB Prediction	59.40	79.76	68.09	-0.85

Table 7.10: The usefulness of individual features in the Hungarian short news corpus in terms of recall, precision and the F-score.

gap between the CROSS and domain adaptation progressively increases with the amount of the data added. The second diagram shows the results obtained when the Szeged Treebank newspaper domain was the target and news was the source. The results got from this model also exceeded the TARGET results when we added more than 2500 target sentences to the training dataset.

In order to examine the effectiveness of each individual feature, we carried out an ablation analysis. That is, for each LVC specific feature, we trained a CRF classifier with all of the features except that one. We then compared the performance to that got with all the features. In the case of Hungarian, the CRF classifier was trained on the Szeged Treebank short news corpus. In the case of English, we performed the ablation study on the CoNLL-2003 corpus. Tables 7.10 and 7.3 tell us how useful the individual features were for each language. The performance scores of the features were compared with that obtained by applying all the features described here.

In the case of Hungarian, **lists of lemmatised LVCs** and the **Suffix** feature were the most useful: the lack of these features led to the lowest result. Part-of-speech-related features were also important, especially the detailed morphological information (**SubPos**). The other features seemed to have a lower impact on the overall results but were still important. In the case of English, the **Syntax**, **Stem** and **VerbalStem** features were the most useful. However, the features **Suffix** and **LVC list** were less effective, but still contributed to the overall performance.

7.4 Discussion

The results of our experiments can be evaluated from different aspects. First, we pay attention to differences between the results obtained by different methods, then to domain differences. Afterwords, we will turn to interlingual differences.

Feature	Recall	Precision	F-score	Difference
Dictionary-lookup	7.65	69.23	13.79	-
Base features	28.39	47.86	35.64	-
All features	50.85	71.43	59.41	-
LVC Lists	48.73	70.12	57.50	-1.91
Prediction List	47.03	68.94	55.92	-3.49
POS-pattern	46.19	70.78	55.90	-3.51
VerbalStem	41.95	68.75	52.11	-7.30
Syntax	40.25	64.63	49.61	-9.8
Lemma	42.37	62.89	50.63	-8.78
Suffix	49.58	72.22	58.79	-0.62
Other entities	46.61	68.75	55.56	-3.85

Table 7.11: The usefulness of individual features in the English CoNLL-2003 corpus in terms of recall, precision and the F-score.

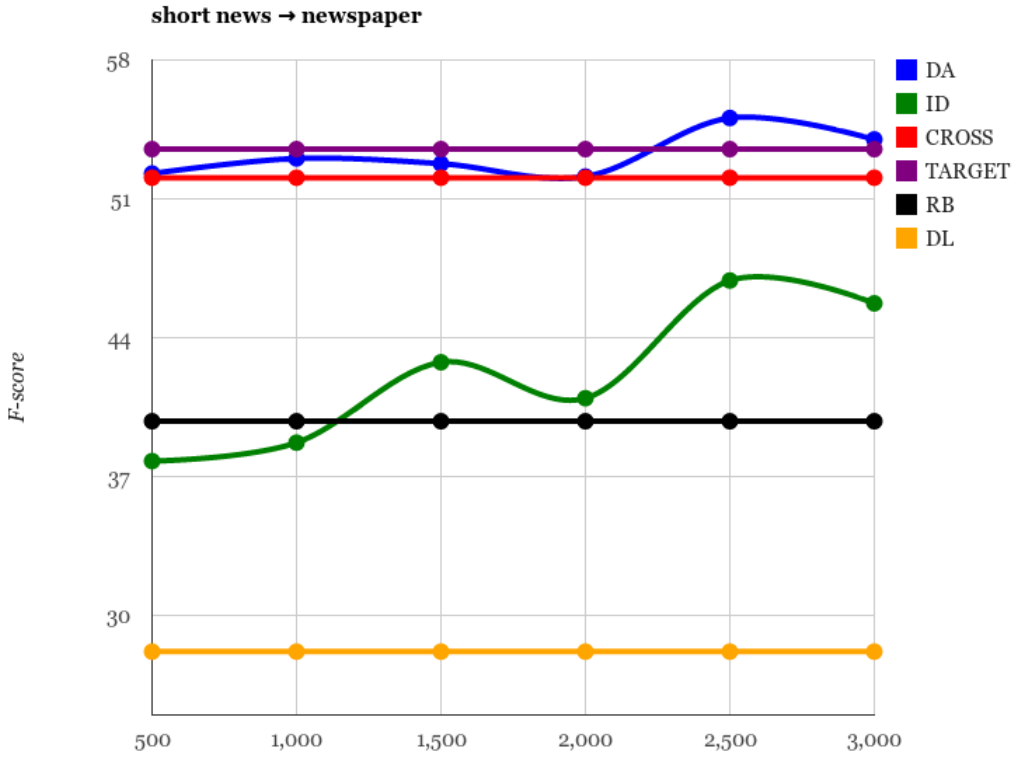

Figure 7.1: The effect of varying the size of the target data on detecting Hungarian LVCs in the newspaper corpus when short news was the source corpus. **DA:** domain adaptation setting. **ID:** training on a limited set of target data. **CROSS:** cross-domain setting. **TARGET:** in-domain setting. RB: rule-based methods. **DL:** dictionary-lookup

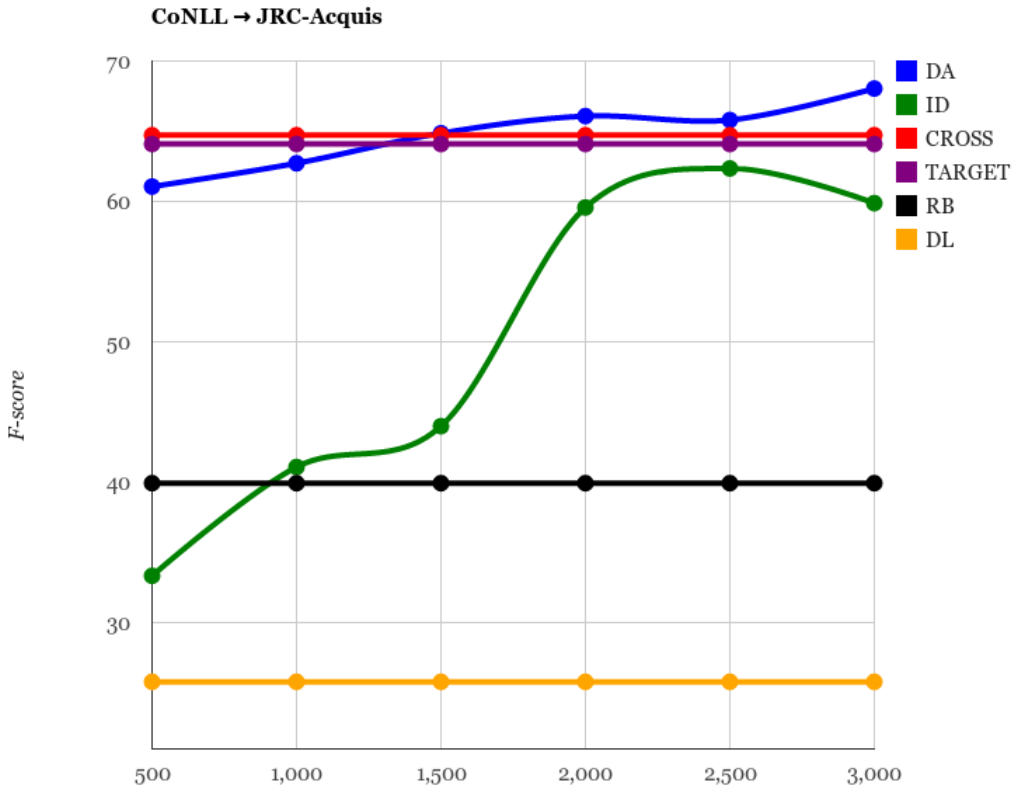

Figure 7.2: The effect of the size of the target data on detecting English LVCs in the JRC-Acquis corpus when the CoNLL dataset was the source corpus. **DA:** domain adaptation setting. **ID:** training on a limited set of target data. **CROSS:** cross-domain setting. **TARGET:** in-domain setting. **RB:** rule-based methods. **DL:** dictionary-lookup.

	SzegedParalellFX			**JRC-Acquis**			**CoNLL-2003**		
Length	**Recall**	**Precision**	**F-score**	**Recall**	**Precision**	**F-score**	**Recall**	**Precision**	**F-score**
2	73.74	86.90	79.78	68.29	84.85	75.68	60.29	77.36	67.77
3	54.67	70.69	61.65	56.19	77.63	65.19	51.55	71.43	59.88
4≤	33.98	58.33	42.94	41.18	63.64	50.00	40.28	64.44	49.57
All	54.29	73.64	62.5	55.38	76.06	64.09	50.85	71.43	59.41

Table 7.12: Results obtained for LVCs with different lengths in terms of recall, precision and F-score on English corpora.

	SzegedParalellFX			JRC-Acquis			CoNLL-2003		
Length	**Recall**	**Precision**	**F-score**	**Recall**	**Precision**	**F-score**	**Recall**	**Precision**	**F-score**
2	48.42	68.62	56.78	76.92	86.71	81.52	66.00	80.39	72.49
3	11.11	33.33	16.67	29.17	53.85	37.84	20.41	62.50	30.77
4≤	0.00	0.00	0.00	18.18	100.00	30.77	16.67	100.00	28.57
All	44.69	66.67	53.51	73.02	85.98	78.97	60.82	79.58	68.94

Table 7.13: Results obtained for LVCs having different word lengths in terms of recall, precision and F-score on Hungarian corpora.

7.4.1 Differences in the Performance of Methods

Machine learning methods applied here consistently outperformed our baseline models (i.e. the rule-based model and dictionary-lookup), which demonstrates that our CRF-based approach can be suitably applied to LVC detection. This is also supported by the fact that our model outperformed that of Tu and Roth (2011) using the same test set. As illustrated by our ablation analysis, the most useful features of the model were morphological, but the effect of syntactic information was more noticeable in English than in Hungarian. Since Hungarian morphology encodes a lot of (morpho)syntactic information, it is not surprising that syntax contributes to LVC detection to a lesser extent in a morphologically rich language although the quality of tagging may also influence the results. Furthermore, the Suffix feature proved more useful for Hungarian than for English. This may be due to the fact that in English, conversion is also a possible linguistic means for deriving a verb from a noun (such as *change*), while nominal derivation is usually executed by adding derivational suffixes to the verb (such as *ajánl* "to offer" – *ajánlat* "offer") in Hungarian, where conversion is almost never applied. Hence, many Hungarian nouns in LVCs end in a derivational suffix, while in English this is only true for vague action verbs, which means that this feature may play a significant role in distinguishing between vague action verbs and true light verb constructions.

7.4.2 Domain Differences

Our cross domain experiments highlighted the domain dependency of detecting LVCs since the cross domain results were always worse than the corresponding indomain (TARGET) results. In spite of this, when there is only a limited amount of target data available, domain adaptation is more effective since the outdomain dataset also contributes to the training process, and training only on the amount of annotated target data (500, 1000 etc. sentences) cannot achieve such outstanding results. There is only one notable exception: the law domain in Hungarian does not seem to profit from outdomain data: it just confuses learning and even with a small amount of annotated target data (around 1500 sentences) it is possible to beat the results of cross training and domain adaptation. This may be explained by the fact that the legal domain apparently has a specific language different from the other domains. The distance between the domains also justifies this fact: the newspaper and short news domains are more similar to each other than any of the others and the legal domain (see Table 7.4).

The special nature of the legal domain is also evident from the baseline results: compared to the other domains, the rule-based system is able to achieve a fairly good result (58.56%). This suggests that the morphological and syntactic patterns of LVCs in the Hungarian law corpus typically follow the canonical form of Hungarian LVCs and thus can be identified by rules.

In English, the effect of using outdomain data is especially fruitful in the case of the short news domain, which may be attributed to the fact that in this domain, the frequency of LVCs is lower than those in the other domains: 4.5% of the sentences contain an LVC, in contrast with the newspaper and law domains (8.67% and 9.08%, respectively). Thus, the same number of target sentences contain fewer LVCs on average and outdomain data may provide some additional training examples. Still, cross-domain results can substantially be improved by adding target data to the newspaper domain, which suggests that this domain has some special characteristics which can be only learned from the target data. In the case of the legal domain, domain adaptation even outperformed results achieved by training exclusively on the target dataset in a 10-fold cross validation setting, which is due to the fact that the legal domain contains the fewest LVCs and also that there is not such a big difference among the domains in English as in Hungarian, where adding outdomain data to the legal domain just confused learning.

Cross-training by itself did not prove sufficient in many cases, so to reduce the gap between domains, the inclusion of annotated target data into the training dataset was necessary. The domain adaptation settings told us that by adding some outdomain data to the training dataset, it was possible to achieve results similar to – or in some cases, even better than – the target results. It was also found (see Figure 7.2) that similar results could be achieved on e.g. the JRC-Acquis corpus if we had (1) 2500 annotated target sentences and a substantial amount of annotated outdomain data or (2) at least 5000 annotated target sentences. These scores are comparable to those reported in Szarvas et al. (2012), where the domain specificity of uncertainty cue detection is analyzed in detail.

The legal domain apparently differs from the other two in both languages. The best TARGET results were achieved on this domain, which may be because this is the most homogeneous domain: the law corpora contain the fewest LVC lemmas and the average frequency of LVC lemmas was the highest here. Moreover, the number of hapax legomena (i.e. LVCs occurring exactly once in the corpus) was low compared with the other corpora. This also explained why it was straightforward to adapt a model to the law domain, whereas it was difficult to adapt a model from it to other domains: the limited legal LVC vocabulary could be effectively learned from a small amount of target data whereas the more extensive vocabulary of the newspaper and short news domains could not be easily acquired if the training dataset contained lots of texts taken from the source domain (i.e. law) and only a few sentences taken from the target domain.

The Hungarian newspaper domain turned out to be the hardest for LVC detection among all corpora, where it achieved a TARGET F-score of just 53.51. This corpus seemed to contain the most heterogeneous LVCs and their distribution is rather balanced; in other words, there were no really frequent LVCs which may be the reason for the big percentage of LVC occurrences. What is more, LVCs with non-typical verbal components frequently occurred

in this corpus, which makes their identification harder (see Section 7.4.4). Lastly, certain errors in LVC detection were simply due to erroneous annotation.

7.4.3 Differences between English and Hungarian Results

Comparing the results obtained for the two languages, it is striking that the Hungarian results are generally better than the English results. This might be due to several factors. First, in Hungarian, datasets were much bigger than those in English, hence the training datasets contained more examples, which probably had a beneficial effect on the results. However, the general proportion of LVCs is not significantly different in the two languages as far as the LVC/verb ratio or LVC/token ratio is concerned. Hence we think that if we had access to more domain-specific data in English, we could achieve better results on the English corpora as well. Second, our feature set included a lot of morphological features, which are especially helpful for a morphologically rich language. Third, shorter LVCs were easier to identify (see Tables 7.12 and 7.13) and about 90% of the Hungarian LVCs are bigrams, which is true only for LVC lemmas in English (see Table 7.15). This is primarily due to language specific rules. On the one hand, in Hungarian, most LVCs do not have an article within the construction, but this is often the case with their English equivalents (cf. *döntést hoz* (decision-ACC brings) vs. *make a decision*). On the other hand, the canonical order of the Hungarian construction is noun + verb, hence modifiers of the noun do not go in between the noun and the verb, but in English, if the noun has premodifiers, they go in between the verb and the noun. Compare:

(7.1) make a very good decision
nagyon jó döntést hoz (very good decision-ACC brings)

In the Hungarian construction, the noun and the verb are adjacent, while in English they are not, which – given that CRF-based approaches are optimized for sequence labeling – results in an easier detectability of Hungarian LVCs.

7.4.4 Error Analysis

In order to gain a deeper understanding of the system's performance, we carried out an error analysis of the data. Besides annotation errors, in many cases, erroneous predictions were related to incorrect POS-tags. In Hungarian, a common error of the POS-tagger was that past tense verbs were often tagged as adjectives (past participles – the word form of which coincides with past tense verbs – do not have a distinct code but are tagged as adjectives), and an adjective + noun sequence was not marked as an LVC. In English, participial occurrences of LVCs were also marked by the system, e.g. *taking a decision* can be a participle form and a verbal form as well, depending on the context. However, we focused only on verbal occurrences and removed participial LVCs from the gold standard data before evaluation, thus if a participial occurrence of an LVC was marked, it was treated as a false positive.

An interesting source of error in Hungarian was related to lemmatization. Some word forms can be ambiguous between the derived forms of two verbal stems: for instance, *vetet* can be a causative form of *vesz* "buy" and *vet* "sow" as well. While *vesz* is a typical light

	English LVCs					
	SzPFX		JRC-Acquis		CoNLL-2003	
Length	**#**	**%**	**#**	**%**	**#**	**%**
2	99	27.97	42	20.59	67	28.51
3	151	42.65	110	53.92	97	41.28
4≤	104	29.38	52	25.49	71	31.21
sum	354	100.00	204	100.00	235	100.00
	English LVCs filtered					
	SzPFX		JRC-Acquis		CoNLL-2003	
Length	**#**	**%**	**#**	**%**	**#**	**%**
2	203	57.34	139	68.14	104	44.26
3	115	32.49	46	22.55	103	43.83
4≤	36	10.17	19	9.31	28	11.91
sum	354	100.00	204	100.00	235	100.00
	Hungarian LVCs					
	SzT newspaper		SzT law		SzT short news	
Length	**#**	**%**	**#**	**%**	**#**	**%**
2	412	90.95	588	93.48	502	89.17
3	27	5.96	23	3.66	49	8.70
4≤	14	3.09	18	2.86	12	2.13
sum	453	100.00	629	100.00	563	100.00

Table 7.14: The length of LVCs in different corpora.

verb in Hungarian, this is not true for *vet*, which rarely occurs in LVCs, hence a false lemma can easily lead to errors in LVC detection.

The length of LVCs can also have an impact on their detection. Table 7.14 includes some statistics on the length of LVCs. A typical example of a two-token LVC is *take care*, one for a three-token long is *take a decision* and a four-token LVC is *come to a conclusion*. In order to minimize the typological differences between the two languages, we also calculated the length of LVCs and LVC lemmas for English with prepositions and articles omitted and it was found that, like Hungarian, most of the LVC lemmas had just two words (see Table 7.15). Tables 7.12 and 7.13 show that the longer the LVC is, the worse the results are likely to be.

Constructions with non-typical nominal components (i.e. those not derived from a verb) are also harder to detect. Furthermore, constructions with rare verbal components are difficult to recognize, which is especially true for Hungarian newspaper texts. There we can find many verbal components which do not occur among the most frequent ones or they form an LVC only with one or two nouns (e.g. *tüzet nyit* (fire-ACC opens) "to open fire" or *búcsút int* (farewell-ACC waves) "to bid farewell"). Therefore we see that, constructions with non-typical nominal or verbal components and infrequent LVCs are the most difficult to recognize.

	English LVC lemmas					
	SzPFX		JRC-Acquis		CoNLL-2003	
Length	#	%	#	%	#	%
2	76	35.19	26	30.59	53	30.64
3	130	60.19	49	57.65	113	65.32
4≤	10	4.63	10	11.76	7	4.05
sum	216	100.00	85	100.00	173	100.00
	English LVC lemmas filtered					
	SzPFX		JRC-Acquis		CoNLL-2003	
Length	#	%	#	%	#	%
2	213	98.61	84	98.82	167	96.53
3	3	1.39	1	1.18	6	3.47
4≤	0	0.00	0	0.00	0	0.00
sum	216	100.00	85	100.00	173	100.00
	Hungarian LVC lemmas					
	SzT newspaper		SzT law		SzT short news	
Length	#	%	#	%	#	%
2	236	98.74	165	98.80	221	93.64
3	2	0.84	2	1.20	15	6.36
4≤	1	0.42	0	0.00	0	0.00
sum	239	100.00	167	100.00	236	100.00

Table 7.15: The length of LVC lemmas with the prepositions and articles removed.

7.5 Summary of thesis results

The main contributions of this chapter can be summarized as follows:

- Here, we addressed a **broader range** of LVCs than previous studies did. In contrast to most of them, we did not just focus on verb-object pairs. Instead, we identified LVCs that contained adpositional complements or nouns in an oblique case.
- We introduced our **conditional random fields**-based state-of-the-art **tool** for detecting LVCs, which makes use of contextual (shallow linguistic) features and it was able to produce satisfactory results for all of the domains and languages used.
- We reported our results for Hungarian and English corpora as well, which allowed us to draw some conclusions on the **multilingual aspects** of LVC detection.
- In our experiments, we made use of three corpora for both languages. The corpora belong to different domains, namely short news, law and newspaper texts. This selection of data made it possible for us to compare the **domain-specific characteristics** of LVC detection in both languages. We reported results for three domains in two languages, and this allowed us to make **cross-lingual comparisons** for each domain.
- We applied **domain adaptation** techniques in order to reduce the distance between domains in a setting where only limited annotated datasets are available for one of the

domains.

In Vincze et al. (2013b), verbal light verb constructions were identified by using a conditional random fields-based tool. The author implemented the machine learning-based method on English and Hungarian, furthermore he applied domain adaptation techniques. He also investigated the effect of simple domain adaptation techniques to reduce the gap between any two domains. The co-authors of the paper were responsible for the linguistic background and the statistical analysis of the corpus data.

Chapter 8

Full-coverage Identification of English and Hungarian Light Verb Constructions

As we showed in Chapter 7, using the Conditional Random Fields-based approach we were able to recognize verbal LVCs in English and Hungarian running texts, but this approach could not handle other types of LVCs like SPLIT and PART. As we described in Section 2.5.2, the noun of the split LVCs may be situated far from the verb in the sentence, so the CRF-based model could not handle it. However, our goal here is to identify each LVC occurrence in running texts, i.e. to take input sentences such as *Where will you deliver your next lecture?* and mark each LVC in it. In this chapter, we focus on the full-coverage identification of light verb constructions. Our basic approach is to syntactically parse each sentence and extract potential LVCs with different candidate extraction methods. Also, we will investigate the performance of different candidate extraction methods on these full-coverage LVC annotated corpora on English and Hungarian, and we will argue that less severe candidate extraction methods should be applied. Afterwards, a binary classification can be used to automatically classify potential LVCs as LVCs or not. For the automatic classification of candidate LVCs, we implement a machine learning approach which is based on a rich feature set. Then, we compare the results of our experiments achieved on the English and Hungarian part of the SzegedParalellFX English–Hungarian parallel corpus, where LVCs were manually annotated in both languages.

8.1 Identification of Restricted Sets of Light Verb Constructions in Earlier Studies

Given the inconsistencies in the literature as regards the coverage of light verb constructions to be identified, we carried out some statistical analyses on the distribution of different types of LVCs in order to justify which approach was the most realistic for identifying English light verb constructions in running texts. We made use of Wiki50 (see Section 3.5) and SzegedParalellFX (see Section 3.7) corpora, in which all types and occurrences of light verb constructions were manually annotated. As earlier studies applied some restrictions on

LVCs, we will show here how these restrictions can affect LVC detection.

8.1.1 Morpho-syntactic Restrictions

Based on the previously used methods (Tu and Roth, 2011; Stevenson et al., 2004; Tan et al., 2006; Fazly and Stevenson, 2007), which just treated the verb-object pairs as potential LVCs, we examined the distribution of dependency label types on the Wiki50 and the SZPFX corpora. Table 8.3 shows that only 73.91% of annotated LVCs on the Wiki50 and 70.6% on the SZPFX had a verb-object syntactic relation. Despite the fact that English verb + prepositional constructions were mostly neglected in previous research (see Section 7.1.2), each corpus contained several examples of structures like *take into consideration* or *come into contact* and the ratio of such LVC lemmas was 11.8% and 9.6% in the English Wiki50 and the English part of SzegedParalellFX corpora, respectively. In addition to the verb + object or verb + prepositional object constructions, there are several other syntactic constructions where LVCs can occur in English due to their syntactic flexibility. For instance, the nominal component can become the subject in a passive sentence (*the **photo** has been **taken***), or it can be extended by a relative clause (*the **photo** that has been **taken***). These cases are responsible for 7.6% and 19.4% of the LVC occurrences in the Wiki50 and the English part of SzegedParalellFX corpora, respectively. These types cannot be identified when only verb + object pairs are used for LVC candidate selection.

8.1.2 Lexical Restrictions

Some researchers filtered light verb construction candidates by selecting only certain verbs that may be part of the construction. One such example is Tu and Roth (2011) (see Section 7.1.2), where the authors chose six light verbs (*make*, *take*, *have*, *give*, *do*, *get*). As the full-covered annotated corpora were available, we were able to check what percentage of light verb constructions could be covered with this selection. Table 8.1 lists the number of the most frequent verbs in the English corpora and in the union of the two English datasets, while Table 8.2 shows the number of the most frequent verbs in the Hungarian corpora.

As can be seen, *make*, *take*, *give* and *have* are among the top 5 light verbs in both corpora. *Do* also frequently occurs, especially in SzegedParallelFX, but *get* does not belong to the most frequent light verbs, at least in these datasets. These six verbs are altogether responsible for about 49% and 63% of all light verb constructions in Wiki50 and the SzegedParalellFX corpora, respectively. In addition, 62 different light verbs occurred in the Wiki50 corpus and 102 in the English part of SzegedParallelFX corpus, respectively. Furthermore, 236 light verbs occurred in Szeged TreebankFX and 180 in the Hungarian part of SzegedParallelFX corpora, respectively. Moreover, the top six verbs (*ad*, *vesz*, *hoz*, *tesz*, *köt*, *kerül*) in the Hungarian corpora only cover 57.84% of all LVCs in the two Hungarian corpora. All this indicates that focusing on a reduced set of light verbs will lead to the exclusion of a considerable number of light verb constructions that occur in free texts.

Wiki50		SzegedParallelFX-EN		Union		
light verb	#	light verb	#	light verb	#	%
take	65	take	270	take	335	19.26
make	61	make	260	make	321	18.45
have	29	give	153	give	173	9.95
hold	28	have	134	have	163	9.38
give	20	hold	58	hold	86	4.95
play	13	play	51	play	64	3.68
commit	10	do	36	do	40	2.3
draw	9	come	26	meet	38	2.19
meet	8	meet	30	come	32	1.84
put	8	catch	21	put	26	1.49
bring	6	put	19	catch	21	1.21
come	6	pay	18	offer	21	1.21
go	6	offer	17	commit	20	1.15
gain	5	provide	16	pay	19	1.1
do	4	keep	14	draw	18	1.04

Table 8.1: The most frequent English verbal components.

8.1.3 Semantic Restrictions

Some papers focus only on the identification of true LVCs, neglecting vague action verbs (Stevenson et al., 2004; Tu and Roth, 2011) (see also Section 2.5.2). However, we cannot see any NLP application that can benefit if such a distinction is made since vague action verbs and true LVCs share those properties that are relevant for natural language processing (e.g. they must be treated as one complex predicate (Vincze, 2012)). We also argue that it is important to separate LVCs and idioms because LVCs are semi-productive and semi-compositional – which may be exploited in applications like machine translation or information extraction –, in contrast to idioms, which have neither feature. Overall, we seek to identify all LVCs (not including idioms) in our study and do not restrict ourselves to specific types of LVCs.

8.2 Syntax-based Detection of Light Verb Constructions

To automatically detect LVCs, we employ a two-stage procedure. First, we identify potential LVC candidates in running texts – we empirically compare various candidate extraction methods –, then we use a machine learning-based classifier that exploits a rich feature set to select LVCs from the candidates. Figure 8.1 outlines the process used to identify each individual LVC in a running text.

For this purpose, we make use of the English–Hungarian parallel corpus SzegedParalellFX (see Section 3.7), where LVCs have been manually annotated. The English Wiki50 (see Section 3.5) and the Hungarian Szeged TreebankFX (see Section 3.6) will also be ap-

Szeged TreebankFX		SzegedParallelFX-HU		Union		
light verb	#	light verb	#	light verb	#	%
ad “give”	1424	vesz “take”	152	ad “give”	1569	19.34
vesz “take”	795	ad “give”	145	vesz “take”	947	11.68
hoz “bring”	775	tesz “make, put”	103	hoz “bring”	844	10.41
tesz “make, put”	468	kerül “get done”	72	tesz “make, put”	571	7.04
köt “bind”	348	hoz “bring”	69	köt “bind”	384	4.74
kerül “get done”	304	tart “hold, keep”	60	kerül “get done”	376	4.63
jut “get”	241	kap “get”	58	jut “get”	262	3.23
tart “hold, kepp”	197	nyújt “offer”	58	tart “hold, keep”	257	3.17
lép “step”	164	áll “stand”	37	nyújt “offer”	204	2.52
nyújt “offer”	146	köt “bind”	36	kap “get”	186	2.29
áll “stand”	145	ér “reach”	25	lép “step”	183	2.26
kap “get”	128	jut “get”	21	áll “stand”	182	2.24
végez “carry out”	111	nyílik “open”	19	végez “carry out”	127	1.57
folytat “execute”	105	lép “step”	19	folytat “execute”	115	1.42
ér “reach”	77	játszik “play”	16	ér “reach”	102	1.26

Table 8.2: The most frequent Hungarian verbal components.

plied here.

8.2.1 Candidate Extraction

As we had English and Hungarian full-coverage LVC annotated corpora where each type and the individual occurrence of a LVC was marked in running texts, we were able to examine the characteristics of LVCs in English and Hungarian running texts, and evaluate and compare the performance the different candidate extraction methods.

Table 8.3 shows the distribution of dependency label types provided by the Bohnet parser (Bohnet, 2010) and Stanford parser (Klein and Manning, 2003) for the English corpora. In

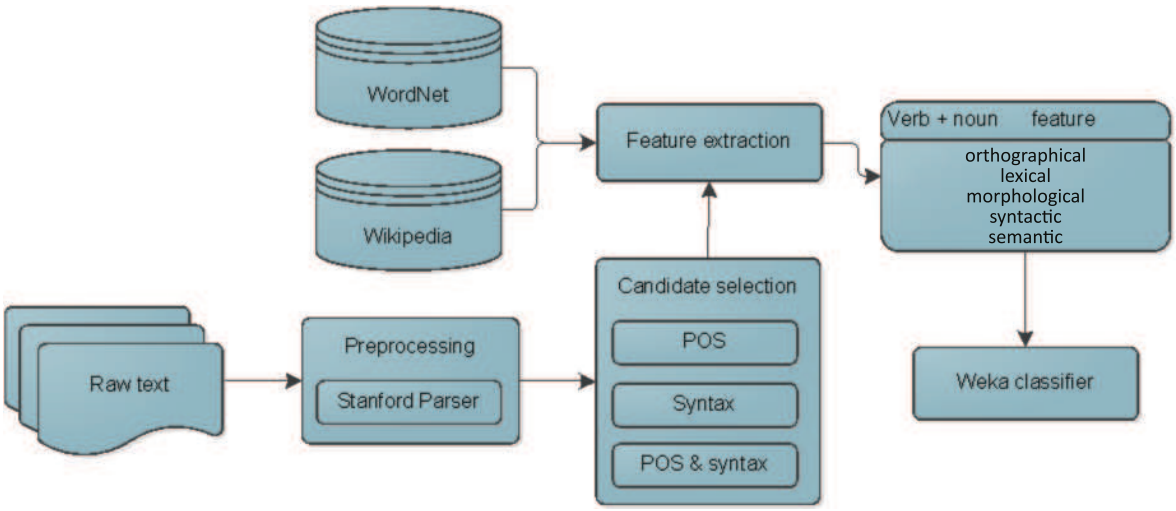

Figure 8.1: System Architecture

order to compare the efficiency of the parsers, both were applied using the same dependency representation. In this phase, we found that the Bohnet parser was more successful, i.e. it could cover more LVCs, hence we applied the Bohnet parser in our remaining experiments. Table 8.4 shows the distribution of dependency label types on Hungarian corpora provided by the magyarlanc parser (Zsibrita et al., 2013), which is the only data-driven dependency parser available for Hungarian.

We defined the extended syntax-based candidate extraction method, where besides the *verb-direct object* dependency relation, other relations were also investigated among verbs and nouns. In the case of English, the *verb-prepositional phrase*, *verb-relative clause*, *noun-participial modifier* and *verb-subject of a passive construction* syntactic relations were also applied, while in Hungarian the *verb-oblique*, *verb-subject* and *noun-attributive* dependency relations were also investigated among the nouns and verbs. Here, 90.76% of LVCs in the Wiki50 and 87.75% in the English part of SzegedParalellFX corpus could be identified by applying the extended syntax-based candidate extraction method, while the corresponding scores were 92.07% and 87.76% on the Szeged TreebankFX and the Hungarian part of SzegedParalellFX, respectively.

It should be added that some rare examples of split LVCs where the nominal component is part of the object, preceded by a quantifying expression like *he **gained** much of his **fame*** cannot easily be identified by syntax-based methods since there is no direct link between the verb and the noun. In other cases, the omission of LVCs from candidates is due to the rare and atypical syntactic relation between the noun and the verb (e.g. `dep` in *reach conform*). Despite this, such cases were also included in the training and evaluation datasets as positive examples.

Our second candidate extractor is the morphology-based candidate extraction method (Nagy T. et al., 2011b), which was also applied for extracting potential LVCs. In this case, a token sequence was treated as a potential LVC if the POS-tag sequence matched one pattern typical of LVCs (e.g. `VERB-NOUN`). Although this method was less effective

Edge type	Wiki50				SzegedParalellFX-EN			
	Stanford		Bohnet		Stanford		Bohnet	
dobj	265	72.01	272	73.91	901	65.71	968	70.6
pobj	43	11.69	43	11.69	93	6.78	93	6.78
nsubjpass	8	2.17	6	1.63	61	4.45	73	5.32
rcmod	5	1.36	6	1.63	30	2.19	38	2.77
partmod	6	1.63	7	1.9	21	1.53	31	2.26
sum	327	88.86	334	90.76	1,106	80.67	1,203	87.75
other	22	4.35	15	4.07	8	0.58	31	2.26
none	25	6.79	19	5.17	257	18.75	137	9.99
sum	368	100.0	368	100.0	1,371	100.0	1,371	100.0

Table 8.3: Edge types in Wiki50 and the English part of SzegedParalellFX corpora. **dobj:** object. **pobj:** preposition. **nsubjpass:** subject of a passive construction. **rcmod:** relative clause. **partmod:** participial modifier. **other:** other dependency labels. **none:** no direct syntactic connection between the verb and noun.

Edge type	Szeged TreebankFX		SzegedParalellFX-HU	
OBJ	2,420	56.01	689	50.01
OBL	884	20.46	312	22.66
SUBJ	210	4.86	88	6.39
ATT	155	3.58	51	3.71
sum	3,669	84.91	1,140	82.79
other	302	6.99	140	10.18
none	350	8.1	97	7.05
sum	4,321	100.0	1,377	100.0

Table 8.4: Edge types in Szeged TreebankFX and the Hungarian part of SzegedParalellFX corpora. **OBJ:** object. **OBL:** oblique. **SUBJ:** subject. **ATT:** attributive. **none:** no direct syntactic connection between the verb and noun.

than the extended syntax-based approach, when we merged the extended syntax-based and morphology-based methods, we were able to identify most of the LVCs in the two English corpora. However, the merged method did not have any noticeable beneficial effect for the Hungarian case.

The authors of Stevenson et al. (2004) and Tu and Roth (2011) filtered LVC candidates by selecting only certain verbs that could be part of the construction, so we checked what percentage of LVCs could be covered with this selection when we treated just the verb-object pairs as LVC candidates. We found that even the least stringent selection covered only 41.88% of the LVCs in Wiki50 and 47.84% in the English part of SzegedParalellFX. Hence, we decided to drop any such constraint.

Tables 8.5 and 8.6 show the results we got by applying the different candidate extraction methods on the English and Hungarian corpora, respectively.

Method	Wiki50 #	Wiki50 %	SzegedParalellFX-EN #	SzegedParalellFX-EN %
Stevenson et al. (2004)	107	29.07	372	27.13
Tu&Roth (2011)	154	41.84	656	47.84
dobj	272	73.91	968	70.6
POS	293	79.61	907	66.15
Syntax	334	90.76	1,203	87.75
POS∪Syntax	339	92.11	1,223	89.2

Table 8.5: The recall of candidate extraction approaches on English corpora. **dobj:** verb-object pairs. **POS:** morphology-based method. **Syntax:** extended syntax-based method. **POS∪Syntactic:** union of the morphology- and extended syntax-based candidate extraction methods.

Method	Szeged TreebankFX #	Szeged TreebankFX %	SzegedParalellFX-HU #	SzegedParalellFX-HU %
obj	2,046	47.35	689	53.04
POS	2,630	60.86	888	68.36
Syntax	3,994	92.43	1,140	87.75
POS∪Syntax	4,015	92.91	1,153	89.2

Table 8.6: The recall of candidate extraction approaches on Hungarian corpora. **obj:** verb-object pairs. **POS:** morphology-based method. **Syntax:** extended syntax-based method. **POS∪Syntactic:** union of the morphology- and extended syntax-based candidate extraction methods.

8.2.2 Candidate Classification

For the automatic classification of the candidate LVCs we implemented a machine learning approach, which we will elaborate upon below. Our method is based on a rich feature set with the following categories: statistical, lexical, morphological, syntactic, orthographic and semantic.

8.2.3 Extended Feature Set

Here, our feature set is based on the features presented in Section 7.2.3. As we treated the LVC detection as a classification problem (as opposed to Chapter 7, where we handle this problem as a sequence labeling problem), we can define some new features.

Orthographic features: The **number of words** of the candidate LVC was also noted and applied as a feature.

Lexical features: Here we also applied lists of lemmatised LVCs, as presented in Section 7.2.3. When we trained our model on Wiki50, the lists were collected from the English part of SzegedParalellFX and vice versa. Similarly, when we trained our model on the Hungarian part of SzegedParalellFX, the lists were collected from the subcorpora of Szeged TreebankFX and the other way around.

Morphological features: The morphological features presented in Section 7.2.3 were

also extended. The English **auxiliaries**, *do* and *have* often occur as light verbs, hence we defined a feature for the two verbs to denote whether or not they were auxiliaries in a given sentence. In addition, the Hungarian LVC candidates which contain the copula *van* "be" were filtered. The POS code of the next word of LVC candidate was also applied as a feature. As Hungarian is a morphologically rich language, we were able to define various new morphology-based features. For this, we simply used the **morphological codes** of words. In this way, we defined some agglutinative features like the mood of the verbs (`Mood`) and the type (`SubPos`), the case (`Cas`), the number of possessor (`NumP`), the person of the possessor `PerP` and the number of the possessed `NumPd` of the noun. Nouns which were historically derived from verbs but were not treated as derivation by the Hungarian morphological parser were also added as a feature.

Syntactic features: As the candidate extraction methods basically depended on the **dependency relation** between the noun and the verb, they could also be utilised in identifying LVCs. Though the `dobj`, `prep`, `rcmod`, `partmod` or `nsubjpass` dependency labels were used in candidate extraction in the case of English, these syntactic relations were defined as features, while the `att`, `obj`, `obl`, `subj` dependency relations were used in the case of Hungarian. When the noun had a **determiner** in the candidate LVC, it was also encoded as another syntactic feature.

Semantic features: This feature also exploited the fact that the nominal component is usually derived from verbs. Consequently, the `activity` or `event` semantic senses were looked for among the upper level hyperonyms of the head of the noun phrase in English WordNet 3.1[1] and in the Hungarian WordNet (Miháltz et al., 2008). In the case of English, `activity` or `event` semantic senses were looked for among the upper level hyperonyms of the head of the noun phrase in WordNet 3.1 (Fellbaum, 1998), while in Hungarian, `tevékenység` and `esemény` were looked for in the Hungarian WordNet (Miháltz et al., 2008).

Our feature set includes language-independent and language-specific features as well. Language-independent features can be used to acquire general features of LVCs, while language-specific features can be applied due to the different grammatical characteristics of the two languages or due to the availability of different resources. Table 8.7 shows which features were applied for which language.

8.2.4 Machine Learning Based Candidate Classification

We experimented with several learning algorithms and our preliminary results showed that decision trees performed the best. This is probably due to the fact that our feature set consisted of a few compact – i.e. high-level – features. We trained the J48 classifier of the WEKA package (Hall et al., 2009), which implements the decision trees algorithm C4.5 (Quinlan, 1993) with the above-mentioned feature set. We provide results with Support Vector Machines (SVM) (Cortes and Vapnik, 1995) as well, to compare the performance of our methods with Tu and Roth (2011). In the case of Hungarian, the J48 decision tree was trained with the Hungarian-specific LVC feature set and was evaluated on the Hungarian corpora.

As the corpora in question were not sufficiently big for splitting into training and test sets

[1] goo.gl/lXLZWQ

Features	Base	English	Hungarian
Orthographical	•	–	–
VerbalStem	•	–	–
POS pattern	•	–	–
LVC list	•	–	–
Light verb list	•	–	–
Semantic features	•	–	–
Syntax features	•	–	–
Auxiliary verb	–	•	–
Determiner	–	•	–
Noun list	–	•	–
POS After	–	•	–
LVC frequency statistics	–	•	–
Agglutinative morphology	–	–	•
Historical derivation	–	–	•

Table 8.7: The basic feature set and language-specific features.

of appropriate size, besides, the different annotation principles ruled out the possibility of expanding the training sets with another corpus, we evaluated our models in a 10-fold cross validation manner on the English and Hungarian corpora and the Tu&Roth dataset. In order to compare our method's performance with the Tu&Roth approach, we used an accuracy score as an evaluation metric on the Tu&Roth dataset, where positive and negative examples were also marked and the different examples were balanced.

In the case of Wiki50, SzegedParalellFX and Szeged TreebankFX, where only the positive LVCs were annotated, we employed $F_{\beta=1}$ scores interpreted on the positive class as an evaluation metric. Moreover, we treated all potential LVCs as negative which were extracted by different extraction methods, but were not marked as positive in the gold standard. The resulting datasets were not balanced and the number of negative examples basically depended on the candidate extraction method applied.

However, some positive elements in the corpora were not covered in the candidate classification phase, as the candidate extraction methods applied could not detect all LVCs in the corpus data. We treated the omitted LVCs as false negatives in our evaluation.

8.3 Results

As a baseline, a context-free dictionary lookup method was applied on both languages. The same lemmatised LVC lists were used as those in the lexical features, described in Section 8.2.3. We marked candidates of the union of the extended syntax-based and morphology-based methods as LVCs if the candidate light verb and one of its syntactic dependents were found on the list.

8.3.1 Results on English Corpora

Tables 8.8 and 8.9 list the results got on the Wiki50 and the English part of SzegedParalellFX corpora using the baseline dictionary lookup and our machine learning approach with different machine learning algorithms and different candidate extraction methods. The dictionary lookup approach got the highest precision on the English part of SzegedParalellFX, namely 72.65%. However, the machine learning-based approach proved to be the most successful as it achieved an F-score that was 19.69 higher than that with dictionary lookup. Hence, this method turned out to be more effective regarding recall.

	Wiki50					
	J48			**SVM**		
Method	**Recall**	**Precision**	**F-score**	**Recall**	**Precision**	**F-score**
DL	36.26	56.11	44.05	36.26	56.11	44.05
POS	46.2	60.65	52.45	48.64	54.1	51.23
Syntax	47.55	61.29	53.55	51.63	50.99	51.31
POS∪Syntax	51.09	58.99	54.76	51.36	49.72	50.52

Table 8.8: Results obtained in terms of recall, precision and F-score on the Wiki50 corpus. **DL:** dictionary lookup. **POS:** morphology-based candidate extraction. **Syntax:** extended syntax-based candidate extraction. **POS∪Syntax:** the merged set of the morphology-based and syntax-based candidate extraction methods.

	SzegedParalellFX-EN					
	J48			**SVM**		
Method	**Recall**	**Precision**	**F-score**	**Recall**	**Precision**	**F-score**
DL	27.83	72.65	40.24	27.83	72.65	40.24
POS	43.02	66.12	52.12	42.42	54.88	47.85
Syntax	56.17	63.25	59.5	54.03	54.38	54.2
POS∪Syntax	56.91	63.29	59.93	55.14	55.84	55.49

Table 8.9: Results obtained in terms of recall, precision and F-score on the English part of the SzegedParalellFX corpus. **DL:** dictionary lookup. **POS:** morphology-based candidate extraction. **Syntax:** extended syntax-based candidate extraction. **POS∪Syntax:** the merged set of the morphology-based and syntax-based candidate extraction methods.

Our machine learning-based approach with different candidate extraction methods demonstrated a consistent performance (i.e. an F-score over 50) on the Wiki50 and the English part of SzegedParalellFX corpora. It is also seen that our machine learning approach with the union of the morphology- and extended syntax-based candidate extraction methods is the most successful method in the case of Wiki50 and the English part of SzegedParalellFX. On both corpora, it achieved an F-score that was higher than that of the dictionary lookup approach (the difference being 10 and 19 percentage points in the case of Wiki50 and the English part of SzegedParalellFX, respectively).

	Szeged TreebankFX					
	J48			SVM		
Method	**Recall**	**Precision**	**F-score**	**Recall**	**Precision**	**F-score**
DL	20.99	53.28	30.02	20.99	53.28	30.02
POS	38.88	68.24	49.54	48.64	54.1	51.23
Syntax	52.18	67.68	58.92	51.63	50.99	51.31
POS∪Syntax	51.09	58.99	54.76	51.36	49.72	50.52

Table 8.10: Results obtained in terms of recall, precision and F-score on Szeged TreebankFX. **DL:** dictionary lookup. **POS:** morphology-based candidate extraction. **Syntax:** extended syntax-based candidate extraction. **POS∪Syntax:** the merged set of the morphology-based and syntax-based candidate extraction methods.

	SzegedParalellFX-HU					
	J48			SVM		
Method	**Recall**	**Precision**	**F-score**	**Recall**	**Precision**	**F-score**
DL	34.46	63.24	44.59	34.46	63.24	44.59
POS	40.95	68.54	51.27	25.79	65.37	36.99
Syntax	50.04	66.1	56.96	26.02	65.11	37.18
POS∪Syntax	49.04	67.2	56.7	30.72	59.16	40.44

Table 8.11: Results obtained in terms of recall, precision and F-score on the Hungarian part of SzegedParalellFX corpus. **DL:** dictionary lookup. **POS:** morphology-based candidate extraction. **Syntax:** extended syntax-based candidate extraction. **POS∪Syntax:** the merged set of the morphology-based and syntax-based candidate extraction methods.

8.3.2 Results on Hungarian Corpora

Tables 8.10 and 8.11 list the results achieved by different methods on Hungarian corpora. The baseline dictionary lookup method was more successful on the Hungarian part of SzegedParalellFX than on Szeged TreebankFX. It achieved F-scores of 44.59, and 30.02 on Szeged TreebankFX, respectively. Moreover, the machine learning based method yielded approximately the same F-scores (i.e. an F-score over 56) on the two Hungarian corpora. Similar to the English results, the machine learning based approach achieved an F-score that was higher than that of the dictionary lookup method on both Hungarian corpora.

As SzegedParalellFX is a parallel corpus, we were able to compare the results on the two different languages. Table 8.12 lists the results got on the two different parts of SzegedParalellFX using the machine learning-based and the baseline dictionary lookup approaches. The dictionary lookup approach yielded the highest precision on the English part of SzegedParalellFX, namely 73.71%. At the same time, the machine learning and dictionary lookup methods got roughly the same precision score on the Hungarian part of SzegedParalellFX, but again the machine learning-based approach achieved the best F-score. Also, in the case of English the dictionary lookup method got a higher precision score, but the machine learning approach proved to be more effective.

	SzegedParalellFX					
	English			Hungarian		
Method	**Recall**	**Precision**	**F-score**	**Recall**	**Precision**	**F-score**
DL	29.22	73.71	41.67	34.46	63.24	44.59
ML	56.91	63.29	59.93	50.04	66.1	56.96

Table 8.12: Results obtained in terms of recall, precision and F-score for the SzegedParalellFX corpus. **DL:** dictionary lookup method. **ML:** machine learning approach.

8.3.3 Results on the Tu&Roth Dataset

In order to compare the performance of our system with others, we evaluated it – with the necessary modifications (e.g. detecting only true light verbs, considering only six verbs and just the `dobj` syntactic relation was used during the candidate extraction phase) – on the Tu&Roth dataset (Tu and Roth, 2011) too. As Table 8.13 indicates, the dictionary lookup method was less effective on the Tu&Roth dataset. Due to limitations of the size of the dictionary applied, this method yielded the highest F-score on the negative class.

Method	**Accuracy**	**F1+**	**F1-**
DL	61.25	56.96	64.76
Tu&Roth Original	68.52	**75.36**	56.41
ML	**72.51**	74.73	**70.5**

Table 8.13: Results got from applying different methods on the Tu&Roth dataset. **DL:** dictionary lookup. **Tu&Roth Original:** the results of Tu & Roth (2011). **ML:** our machine learning-based model.

8.3.4 Ablation Analysis

To examine the effectiveness of each individual feature of the machine learning based candidate classification, we carried out an ablation analysis. For each feature type, we trained a J48 classifier with all of the features except that one. We then compared the performance to that got with all the features. We also investigated how language-specific features improved the performance compared to the base feature set. We then compared the performance to that got with all the features. Table 8.14 shows the contribution of each individual feature type on the SzegedParalellFX corpus.

8.4 Discussion

After presenting the methods used and results obtained, we will now discuss our main findings on LVC detection.

	English				Hungarian			
Feature	**Recall**	**Precision**	**F-score**	**Diff**	**Recall**	**Precision**	**F-score**	**Diff**
All	**56.91**	**63.29**	**59.93**	**–**	**50.04**	**66.1**	**56.96**	**–**
Lexical	28.6	71.24	40.82	-19.11	34.33	57.21	42.91	-14.05
Morphological	62.3	54.77	58.29	-1.64	49.42	62.54	55.21	-1.75
Orthographic	55.95	63.54	59.5	-0.43	48.58	59.93	53.66	-3.3
Syntactic	55.88	60.49	58.09	-1.84	48.34	65.63	55.68	-1.28
Semantic	54.55	61.38	57.76	-2.17	65.51	49.87	56.63	-0.33
Statistical	55.51	60.07	57.7	-2.23	–	–	–	–
Language-specific	55.88	60.49	58.09	-1.84	49.04	63.51	55.34	-1.62

Table 8.14: The usefulness of individual features in terms of precision, recall and F-score using the SzegedParalellFX corpus.

8.4.1 Candidate Extraction

The machine learning-based method applied easily outperformed our dictionary lookup baseline model on both languages and all corpora, which underlines the fact that our approach can be suitably applied to LVC detection. As Tables 8.8, 8.9, 8.10 and 8.11 show, our presented method proved to be the most robust as it was able to achieve roughly the same recall, precision and F-score on the English and Hungarian corpora. Our system's performance primarily depends on the applied candidate extraction method. However, the recall score is the main difference among the three different approaches applied when we evaluated them on the Wiki50 and SZPFX corpora. It is also due to the candidate extraction methods applied, which covered different sets of the LVCs in the corpora. The number of extracted candidate examples basically depended on the candidate extraction method applied. In other words, less severe candidate extraction methods were able to cover more LVCs in the corpus data. At the same time they may extract candidates that are difficult (or more difficult) to classify, which could influence the precision score. In the case of dictionary lookup, a higher recall score was primarily limited by the size of the dictionary, but this method managed to achieve a fairly good precision score.

8.4.2 Features

From our results, we see that our base system is robust enough to produce about the same performance on two typologically different languages. Language-specific features further contribute to the performance, as shown by our own ablation analysis. It should also be mentioned that some of the base features (e.g. POS-patterns, which we thought would be useful for English due to the fixed word order) were originally inspired by one of the languages and later expanded to the other one (i.e. they were included in the base feature set) since it was also effective in the case of the other language. Thus, a multilingual approach may be also beneficial in the case of monolingual applications as well.

As for the effectiveness of morphological and syntactic features, morphological features perform better on a language with a rich morphological representation (Hungarian). How-

ever, syntax plays a more important role in LVC detection in English: the added value of syntax is higher for the English corpora than for the Hungarian one, where syntactic features are also encoded in suffixes, i.e. morphological information.

As our ablation analysis revealed, each type of feature contributed to the overall performance. The most important feature in our system is the list of the most frequent light verbs. The most common verbs in a language are used very frequently in different contexts, with several argument structures and this may lead to the bleaching (or at least generalization) of its semantic content (Altmann, 2005). From this perspective, it is linguistically plausible that the most frequent verbs in a language largely coincide with the most typical light verbs since light verbs lose their original meaning to some extent (see e.g. Sanromán Vilas (2009)).

Besides our ablation analysis, we also examined the decision tree model produced by our experiments. Similar to the results of our ablation analysis we found that the lexical features were the most powerful, the semantic, syntactic and orthographical features were also useful; while statistical and morphological features were less effective but were still exploited by the model.

8.4.3 Comparison of Languages

The most obvious difference between the performances on the two languages is the recall scores (the difference between the two languages on the SzegedParalellFX corpus being 6.87 percentage points). This may be related to the fact that the distribution of light verbs is quite different in the two languages. While the top 15 verbs cover more than 80% of the English LVCs, in Hungarian, this number is only 63% (and in order to reach the same coverage, 38 verbs should be included). Another difference is that there are 102 different verbs in English, which follow the Zipf distribution, on the other hand, there are 157 Hungarian verbs with a more balanced distributional pattern. Thus, fewer verbs cover a greater part of LVCs in English than in Hungarian and this also explains why lexical features contribute more to the overall performance in English. This fact also indicates that if verb lists are further extended, still better recall scores can probably be achieved for both languages.

Since the Tu&Roth dataset was created by collecting sentences that contain verb-object pairs with specific verbs, this dataset contains a lot of negative and ambiguous examples besides annotated LVCs; hence the distribution of LVCs in the Tu&Roth dataset is not comparable to those in Wiki50 or the English part of SzegedParalellFX. In this dataset, only one positive or negative example was annotated in each sentence, and they examined just the verb-object pairs formed with the six verbs as a potential LVC. However, the corpus probably contains other LVCs that were not annotated. For example, in the sentence *it have (sic!) been held that a gift to a charity of shares in a close company* ***gave rise*** *to a charge to capital transfer tax where the company* ***had an interest*** *in possession in a trust*, the phrase *give rise* was listed as a negative example in the Tu&Roth dataset, but *have an interest*, which is another LVC, was not marked either positive or negative. This is problematic if we would like to evaluate our candidate extractor on this dataset since it would identify this phrase, even if it is restricted to verb-object pairs containing one of the six verbs mentioned above, thus yielding false positives already in the candidate extraction phase.

Moreover, the results got with our machine learning approach overperformed those reported in Tu and Roth (2011). This may be attributed to the inclusion of a rich feature set with new features like semantic or morphological features that was used in our system, which demonstrated a consistent performance on the positive and negative classes too.

Comparing the results on the three English corpora, it is salient that the F-score got from applying the methods on the Tu&Roth dataset was considerably better than those got on the other two corpora. This can be explained if we recall that this dataset applies a restricted definition of LVCs, works with only verb-object pairs and, furthermore, it contains constructions with only six light verbs. However, Wiki50 and the English part of SzegedParalellFX contain all LVCs, they include verb + preposition + noun combinations as well, and they are not restricted to six verbs. All these characteristics demonstrate that identifying LVCs in the latter two corpora is a more realistic and challenging task than identifying them in the artificial Tu&Roth dataset. For example, the very frequent and important LVCs like *make a decision*, which was one of the most frequent LVCs in the two full-coverage LVC annotated corpora, are ignored if we only focus on identifying true LVCs. It could be detrimental when a higher level NLP application exploits the LVC detector. For example, when a machine translation application tries to translate the sentence *This question will give rise to many problems unless you make a decision on it*, it could mistranslate it if only the true LVC phrase *give rise* is detected by the LVC detector while the other LVC *make a decision* is not marked. As a consequence, the results got from applying the methods on the Tu&Roth dataset were higher, but we mention that this dataset only focused on a small part of the problem.

8.4.4 Error Analysis

We carried out an error analysis in order to see how our system could be further improved and the errors reduced. We concluded that there were some general and language-specific errors as well. Among the general errors, we found that in the candidate extraction step, it is primarily POS-tagging or parsing errors that result in the omission of certain LVC candidates. In other cases, as Tables 8.3 and 8.4 show, the dependency relation between the nominal and verbal component is missing (recall the example of objects with quantifiers) or it is an atypical one (e.g. `dep`) not included in our list. The lower recall in the case of the English part of SzegedParalellFX can be attributed to the fact that this corpus contains more instances of nominal occurrences of LVCs (e.g. *decision-making* or *record holder*) than Wiki50, which were annotated in the corpora but our morphology-based and extended syntax-based methods were not specifically trained for them since adding POS-patterns like `NOUN-NOUN` or the corresponding syntactic relations would have resulted in the unnecessary inclusion of many nominal compounds. Moreover, we found that LVCs with a rare light verb were difficult to recognize (e.g. *to utter a lie*). In other cases, an originally deverbal noun was used in a lexicalised sense together with a typical light verb (e.g. *buildings are given (something)*) and these candidates were falsely classed as LVCs.

As for the errors made during classification, it seems that it was hard for the classifier to label longer constructions properly. It was especially true when the LVC occurred in a non-canonical form, as in a relative clause (*counterargument that can be made*). Construc-

tions with atypical light verbs (e.g. *cast a glance*) were also somewhat more difficult to find. Nevertheless, some false positives were due to annotation errors in the corpora. A further source of error was that some literal and productive structures like *to give a book (to someone)* – which contains one of the most typical light verbs and the noun is homonymous with the verb *book* "to reserve" – are very difficult to distinguish from LVCs and were in turn marked as LVCs. Moreover, the classification of idioms with a syntactic or morphological structure similar to typical LVCs – *to have a crush on someone* (to be fond of someone), which consists of a typical light verb and a deverbal noun – was also not straightforward.

As for language-specific errors, English verb-particle combinations (VPCs) followed by a noun were often labeled as LVCs such as *make up his mind* or *give in his notice*. Since Wiki50 contains annotated examples for both types of MWEs, the classification of verb + particle/preposition + noun combinations as verb-particle combinations, LVCs or simple verb + prepositional phrase combinations could be a possible direction for future work. In Hungarian, verb + proper noun constructions (*Hamletet játsszák* (Hamlet-ACC play-3PL.DEF) "they are playing Hamlet") were sometimes regarded as LVCs since the morphological analysis does not make a distinction between proper and common nouns. These language-specific errors may be eliminated by integrating a VPC detector and a Named Entity Recognition system into the English and Hungarian systems, respectively.

8.5 Comparison of Sequence Labeling and Full-coverage Identification

We applied two different methods for the automatic detection of LVCs. The main advantage of the full-coverage identification method is that it can handle non-contiguous LVCs and can focus on identifying all types of LVCs. However, this method requires syntactic parsing as it is based on our syntax-based candidate extraction method. Therefore, the quality of the parsing is of primary influence on performance of the full coverage identification method. When a syntactic parser is not available on a special domain (like medical domain) or language (like Punjabi), the usage of the full-coverage identification method is limited. In contrast, the sequence labeling method can only identify contiguous LVCs in raw texts as it uses CRF and treats the automatic identification of LVCs as a sequence labeling problem. The CRF-based method also uses syntactic information as feature, but it can be omitted if precise syntactic parsing is not available. When a precise syntactic parser is available for the actual domain, the full-coverage identification can perform better, but in other cases the usage of the sequence labeling method is recommended.

8.6 Summary of thesis results

The main contributions of this chapter can be summarized as follows:

- We introduced and evaluated systems for **identifying all LVCs and all individual LVC occurrences** in English and Hungarian running texts and we did not restrict

ourselves to certain specific types of LVCs.

- We systematically **compared and evaluated different candidate extraction methods** (earlier published methods and new solutions implemented by us).
- We defined and evaluated several **new feature templates** like semantic or morphological features to select LVCs in context from extracted candidates. For each of the two languages, each type of feature contributed to the overall performance.
- We applied both **language independent and language specific features**: we compared whether the same set of features could be used for both languages, then investigated the benefits of integrating language specific features into the systems and we explored how the systems could be further improved.
- The method proved to be sufficiently robust as it achieved **approximately the same scores on two typologically different languages**.

In Nagy T. et al. (2013), a system was introduced that enables the full coverage identification of English LVCs in running texts. The author implemented the machine learning based method, he added some new features and developed syntax-based candidate extraction methods, however, experimental results are treated as a shared contribution of all authors. The co-authors were responsible for the linguistic background and the idea of the full-coverage identification of LVCs. In Vincze et al. (2013a), Hungarian and English LVCs were identified in free texts. The author contrasted the performance of the applied methods and applied language-specific features on these typologically different languages. The co-authors were responsible for the linguistic background and the interlingual comparisons.

Chapter 9

Summary

9.1 Summary in English

The chief aim of this thesis was to develop machine learning-based approaches to automatically detect different types of multiword expressions in English and Hungarian natural language texts. In our investigations, we paid attention to the characteristics of different types of multiword expressions. As part of this, we implemented novel machine learning-based methods to automatically detect the different types of multiword expressions. The results were experimentally examined and discussed.

First, we presented the background information on multiword expressions, the corpora used and the basics of machine learning. Then we showed how different types of multiword expressions could be detected in natural language texts via automatic methods. Now we will summarize the most important achievements described in the thesis.

9.1.1 Nominal Compound Detection with Wikipedia-Based Methods

In order to automatically identify nominal compounds in raw English texts, dictionary and machine learning-based approaches were applied on different corpora. These approaches made intensive use of Wikipedia data. We also showed how previously identified nominal compounds affect Named Entity Recognition (NER) and vice versa, how nominal compound detection is supported by identified named entities. We found that a prior knowledge of nominal compounds can enhance NER, while previously identified NEs can assist the nominal compound identification process. We also examined the effectiveness of the machine learning-based method when it was trained on an automatically generated silver standard corpus and we demonstrated that this approach can also provide acceptable results. Moreover, we investigated how the size of an automatically generated silver standard corpus can affect the performance of our machine learning-based method. The results we obtained demonstrate that the bigger the dataset, the better the performance should be **(Thesis 1)**.

9.1.2 Named Entity Recognition Problems in Web Mining

We consider named entities to be similar to nominal compounds as they form one semantic unit, consist of more than one word and they function as a noun. Therefore a similar approach was applied for their recognition as in the case of nominal compounds. We focused on Web Mining-related Named Entity Recognition problems like Researcher Affiliation Extraction, Person Attribute Extraction and Company Contact Information Extraction. Since webpages usually contain several noisy and misleading elements (such as menu elements and ads), these can seriously inhibit the proper functioning of NLP tools, so we applied various methods that normalise the content of webpages to automatically detect named entities. In the first step, we focused on the raw textual parts of the webpages, as we found that most of the useful information is available in natural text format in webpages. Here, we automatically detected the relevant sections from the webpages. Afterwards, named entities were automatically detected by applying machine learning-based models. Finally, we validated candidate named entities using application-specific rule-based methods **(Thesis 2)**.

9.1.3 Sequence Labeling for Detecting English and Hungarian Light Verb Constructions

We presented our sequence labelling-based tool developed for identifying verbal light verb constructions in running texts. The flexibility of the tool was demonstrated on two, typologically different languages, namely English and Hungarian. Furthermore, different types of texts may contain different types of light verb constructions, and the frequency of light verb constructions may differ from domain to domain. Therefore, we focused on the portability of models trained on different corpora and we also investigated the effect of simple domain adaptation techniques to reduce the gap between the domains. Our results showed that in spite of their special domain characteristics, out-domain data can also contribute to successful LVC detection in different domains **(Thesis 3)**.

9.1.4 Full-coverage Identification of English and Hungarian Light Verb Constructions

The CRF-based model was able to automatically detect English and Hungarian verbal LVCs, but this approach could not handle other types of LVCs like split light verb constructions (e.g. *a contract which has been recently made*) and participial form (e.g. *contracts made*). Therefore, we focused on the full-coverage identification of light verb constructions. Our presented approach syntactically parsed each sentence and then extracted potential LVCs using different candidate extraction methods. In addition, we investigated the performance of different candidate extraction methods on full-coverage LVC annotated corpora in English and Hungarian, where we found that less severe candidate extraction methods should be applied. Then we followed a machine learning approach that made use of an extended and rich feature set to select LVCs among extracted candidates. The applied method proved to be sufficiently robust as it achieved approximately the same scores on two typologically

different languages **(Thesis 4)**.

9.1.5 Conclusions and Future Work

In this thesis, we focused on the automatic detection of multiword expressions in natural language texts. On the basis of the main contributions, we can argue that:

- Supervised machine learning methods can be successfully applied for the automatic detection of different types of multiword expressions in natural language texts.
- Machine learning-based multiword expression detection can be successfully carried out for English as well as for Hungarian.
- Our supervised machine learning-based model was successfully applied to the automatic detection of nominal compounds from English raw texts.
- We developed a Wikipedia-based dictionary labeling method to automatically detect English nominal compounds.
- A prior knowledge of nominal compounds can enhance Named Entity Recognition, while previously identified named entities can assist the nominal compound identification process.
- The machine learning-based method can also provide acceptable results when it was trained on an automatically generated silver standard corpus.
- As named entities form one semantic unit and may consist of more than one word and function as a noun, we can treat them in a similar way to nominal compounds.
- Our sequence labelling-based tool can be successfully applied for identifying verbal light verb constructions in two typologically different languages, namely English and Hungarian.
- Domain adaptation techniques may help diminish the distance between domains in the automatic detection of light verb constructions.
- Our syntax-based method can be successfully applied for the full-coverage identification of light verb constructions. As a first step, a data-driven candidate extraction method can be utilized. After, a machine learning approach that makes use of an extended and rich feature set selects LVCs among extracted candidates.
- When a precise syntactic parser is available for the actual domain, the full-coverage identification can be performed better. In other cases, the usage of the sequence labeling method is recommended.

Along with the above points, we think that the results of the thesis should be applicable in other areas of NLP research as well as in other disciplines. In several natural language

processing applications like information extraction and retrieval, terminology extraction, machine translation and document classification, it is necessary to identify multiword expressions in context. For example, in machine translation we must know that MWEs form one semantic unit, hence their parts should not be translated separately. For this, MWEs should be identified first in the text to be translated. Information retrieval may also be enhanced by detecting multiword expressions. In another example, LVCs denote one event and again they should be treated as one unit in event extraction tasks. And as before, LVCs in the text must be identified prior to the extraction of events.

In the future, we would like to improve our systems by conducting a detailed analysis of the effect of each feature. We also plan to adapt our tools to other types of multiword expressions like verb-particle constructions and conduct further experiments on languages other than English and Hungarian. In addition, we can improve the methods applied in each language and for each type of MWE by implementing other language-specific features as well. We would also like to provide a standardized (i.e. language-independent) representation of different types of multiword expressions that can be used in machine learning experiments in a language-independent context.

We believe that the fruits of our research on the automatic detection of multiword expressions can be successfully exploited in several NLP tasks and hope that they will contribute to the development of novel approaches in many areas of natural language processing.

9.2 Magyar nyelvű összefoglaló

Az értekezés fő célkitűzése különböző összetett kifejezések automatikus azonosítása angol és magyar nyelvű folyó szövegekben. Vizsgálataink során figyelmet fordítottunk a különböző típusú összetett kifejezések sajátosságaira, továbbá új, gépi tanuláson alapuló eljárásokat implementáltunk az összetett kifejezések automatikus azonosítására.

A jelen értekezésben az általunk elért főbb eredményeket foglaltuk össze. Először a különböző típusú összetett kifejezéseket mutattuk be, majd a felhasznált korpuszokat, valamint az alkalmazott gépi tanulási megközelítéseket ismertettük. Ezek után bemutattuk az összetett kifejezések automatikus azonosítására alkalmas különböző módszereinket.

9.2.1 Az értekezés eredményei

Az értekezésben elért főbb eredmények az alábbi pontokban foglalhatók össze.

9.2.2 Angol összetett főnevek azonosítása Wikipedia-alapú módszerekkel

Az összetett főnevek angol nyelvű folyó szövegekben való automatikus azonosításának érdekében szótáron, illetve gépi tanuláson alapuló megközelítéseket egyaránt vizsgáltunk különböző korpuszokon. Ezek a megközelítések nagymértékben támaszkodtak a Wikipediára. Ismertettük, hogyan hatnak az előzetesen azonosított összetett főnevek a névelem-felismerés hatékonyságára, és fordítva: az azonosított névelemek hogyan segítik az összetett főnevek azonosítását. Úgy találtuk, hogy az összetett főnevek előzetes ismerete javítja a névelem-felismerést, míg a névelemek azonosítása segítheti az összetett kifejezések azonosítását. Ezenkívül megvizsgáltuk az automatikusan annotált tanítóhalmazon tanított gépi tanulási megközelítés hatékonyságát, és úgy találtuk, hogy ez is elfogadható eredményt képes produkálni.

Emellett megvizsgáltuk, hogyan hat az automatikusan annotált tanítókorpusz mérete a gépitanuló-megközelítés hatékonyságára. A kapott eredmények azt mutatták, hogy a nagyobb tanítóhalmazon tanított modellek jobb eredményt értek el, de a hozzáadott érték folyamatosan csökkent. (**1. tézispont**)

9.2.3 Webbányászat alapú névelem-azonosítási problémák

Mivel a névelemek is egy szemantikai egységet jelölnek, és többnyire főnévként funkcionálnak, valamint több szóból is állhatnak, az összetett főnevekhez hasonlóan kezelhettük őket. Ezért a névelemek automatikus azonosítására az összetett főnevekhez hasonló megközelítéseket alkalmazhatunk. Számos névelem-felismerési problémát ismertettek már, mi itt alapvetően a webbányászathoz köthetőekre fókuszáltunk, mint például kutatók affiliációjának kinyerése, személyes információk kinyerése, és vállalkozások elérhetőségeinek kinyerése, amelyek mind névelem-felismerési problémák.

A weboldalak általában sok zajt is tartalmazhatnak (például menüelemeket vagy hirdetéseket), amelyek jelentősen gátolhatják a különböző számítógépes nyelvészeti eszközök megfelelő működését. Ezért különböző megközelítéseket alkalmaztunk a weboldalak szöveges

tartalmának egységesítésére, hogy kinyerhessük azokból a névelemeket. Első lépésben a honlapok folyószöveges részeire koncentráltunk, mivel úgy találtuk, hogy a hasznos információk legjelentősebb része itt fordul elő leggyakrabban. Ezért automatikusan azonosítottuk a releváns részeit az egyes honlapoknak. Ezután a névelemeket gépitanuló-megközelítéssel automatikusan azonosítottuk a honlapok releváns tartalmaiból. Végül feladatspecifikus szabályalapú megközelítések segítségével validáltuk a kinyert névelemeket. (**2. tézispont**)

9.2.4 Angol és magyar nyelvű félig kompozicionális szerkezetek automatikus azonosítása szekvenciajelölő megközelítéssel

Az igei félig kompozicionális szerkezetek folyószövegekben való azonosítására szekvenciajelölésen alapuló megközelítést implementáltunk. Eredményeinket angol és magyar, két tipológiailag különböző nyelven is ismertettük, ezzel demonstrálva megközelítésünk rugalmasságát.

Mivel a különböző típusú szövegek különböző félig kompozicionális szerkezeteket tartalmazhatnak, valamint ezek előfordulási gyakorisága is eltérő lehet a különböző doméneken, ezért az eltérő korpuszokon tanult modellek hordozhatóságát is megvizsgáltuk. A továbbiakban megvizsgáltuk, hogyan tudják egyszerű doménadaptációs módszerek a különböző domének közti különbségeket áthidalni.

A doménsajátosságok ellenére az eredményeink azt mutatják, hogy a doménen kívüli adat képes segíteni a félig kompozicionális szerkezetek eltérő doméneken való automatikus azonosításában. (**3. tézispont**)

9.2.5 Angol és magyar nyelvű félig kompozicionális szerkezetek teljes halmazának automatikus azonosítása

Ugyan a szekvenciajelölő megközelítés képes automatikusan azonosítani az igei félig kompozicionális szerkezeteket angol és magyar nyelvű folyó szövegekben, ugyanakkor nem képes kezelni az egyéb típusú szerkezeteket, úgymint a nem folytonos (SPLIT) és igeneves (PART) szerkezeteket.

Ezért a félig kompozicionális szerkezetek teljes halmazának azonosítására fókuszáltunk.

Az általunk bemutatott módszer először minden mondatot szintaktikailag elemzett, majd különböző jelöltkiválasztó módszerek segítségével kinyerte a lehetséges félig kompozicionális szerkezeteket. Továbbá, megvizsgáltuk ezen jelöltkiválasztó megközelítések hatékonyságát angol és magyar nyelvű félig kompozicionális szerkezetek esetén is. Ezt követően gazdag jellemzőkészleten tanított gépitanuló-modellek segítségével azonosítottuk a félig kompozicionális szerkezeteket. (**4. tézispont**)

9.2.6 Összegzés és jövőbeli tervek

Az értekezésben összetett kifejezések folyó szövegekben való automatikus azonosításával foglalkoztunk. A legfontosabb eredményeink a következő módon összegezhetők:

- különböző típusú összetett kifejezések automatikus azonosítására sikeresen alkalmaztunk felügyelt gépi tanuláson alapuló megközelítéseket;
- sikeresen alkalmaztunk gépitanuló-megközelítéseket összetett kifejezések automatikus azonosítására angol és magyar nyelven;
- összetett főnevek angol nyelvű folyószövegekben való automatikus azonosításához alkalmazhatók felügyelt gépitanuló-megközelítések és Wikipedián alapuló szabályalapú módszerek;
- a névelemek előzetes ismerete segíti az összetett főnevek automatikus azonosítását, valamint a névelem-felismerést támogatják az előzetesen azonosított összetett főnevek;
- összetett főnevek automatikus azonosítása automatikusan annotált tanítóhalmazon tanított gépi modell segítségével is lehetséges;
- a névelemek automatikus azonosítása az összetett főnevek azonosításához hasonló megközelítéseket kíván, mivel azok hasonló tulajdonságokkal bírnak: a névelemek az összetett főnevekhez hasonlóan egy szemantikai egységet jelölnek, több szóból állhatnak, valamint főnévként funkcionálnak;
- igei félig kompozicionális szerkezetek automatikus angol és magyar nyelvű azonosítása feltételes valószínűségi mezőkön alapuló módszerrel;
- doménadaptációs technikák segítségével csökkenthető a domének közti távolság az angol és magyar nyelvű félig kompozicionális szerkezetek esetében;
- szintaxisalapú megközelítés segítségével a félig kompozicionális szerkezetek teljes halmaza azonosítható;
- abban az esetben, ha az adott doménre elérhető jól működő szintaktikai elemző, akkor a félig kompozicionális szerkezetek automatikus azonosítására a szintaxisalapú megközelítés ajánlott, egyébként a szekvenciajelölésen alapuló módszer.

A fentieken kívül az értekezés eredményeit a számítógépes nyelvészet más területein, illetve más tudományterületeken is hasznosítani lehet. Összetett főnevek kontextusukban való automatikus azonosítása számos számítógépes nyelvészeti alkalmazás számára hasznos lehet, mint például információkinyerés és -visszakeresés, terminológiakinyerés, gépi fordítás vagy dokumentumosztályozás. A gépi fordítás esetében tudnunk kell, hogy egy adott összetett kifejezés egy szemantikai egységet jelöl, ezért részeit nem fordíthatjuk külön-külön. Ezért szükséges az összetett kifejezések automatikus azonosítása az automatikus fordítás előtt. Másrészről a félig kompozicionális szerkezetek automatikus azonosítása eseménykinyerő rendszerek építése során elengedhetetlen lehet, mivel azok gyakran egy eseményt jelölnek, és ezért szükséges egy egységként kezelni azokat.

A jövőben szeretnénk továbbfejleszteni rendszereinket az egyes jellemzők hatásainak részletesebb elemzésével. Szintén tervezzük meglévő módszereink adaptálását más összetett

kifejezések automatikus azonosítására, mint például angol vonzatos igék (phrasal verbs), valamint azok angol és magyar nyelveken túli kiterjesztését.

Továbbá javítani kívánjuk meglévő módszereinket új, nyelvspecifikus jellemzők megvalósításával. Annak érdekében, hogy egy nyelvfüggetlen gépitanuló-megközelítést is létrehozhassunk, a jövőben szeretnénk a meglévő jellemzőket általánosítani.

Véleményünk szerint az értekezésben ismertetett összetett kifejezések automatikus azonosítására szolgáló módszerek jól hasznosíthatók számos számítógépes nyelvészeti feladat megoldása során, valamint újfajta megközelítések kidolgozásában.

References

Alonso Ramos, Margarita. 2004. *Las construcciones con verbo de apoyo*. Visor Libros, Madrid.

Altmann, Gabriel. 2005. Diversification processes. In *Handbook of Quantitative Linguistics*, pp. 646–659, Berlin. de Gruyter.

Anastasiou, Dimitra; Carl, Michael. 2008. A Lexicon of shallow-typed German-English MW-Expressions and a German Corpus of MW-Expressions annotated Sentences. In *Proceedings of the LREC 2008 Workshop Towards a Shared Task for Multiword Expressions (MWE 2008)*, pp. 15–18, Marrakech, Morocco.

Apresjan, Jurij D. 2004. O semantičeskoj nepustote i motivirovannosti glagol'nyx leksičeskix funkcij. *Voprosy jazykoznanija*, (4):3–18.

Artiles, Javier; Gonzalo, Julio; Sekine, Satoshi. 2007. The SemEval-2007 WePS evaluation: establishing a benchmark for the web people search task. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, SemEval '07, pp. 64–69, Stroudsburg, PA, USA. Association for Computational Linguistics.

Artiles, Javier; Gonzalo, Julio; Sekine, Satoshi. 2009. WePS 2 Evaluation Campaign: overview of the Web People Search Clustering Task. In *2nd Web People Search Evaluation Workshop (WePS 2009), 18th WWW Conference*.

Artiles, Javier; Borthwick, Andrew; Gonzalo, Julio; Sekine, Satoshi; Amigó, Enrique. 2010. WePS-3 Evaluation Campaign: Overview of the Web People Search Clustering and Attribute Extraction Task. In *Conference on Multilingual and Multimodal Information Access Evaluation (CLEF)*.

Bannard, Colin. 2007. A measure of syntactic flexibility for automatically identifying multiword expressions in corpora. In *Proceedings of the Workshop on a Broader Perspective on Multiword Expressions*, MWE '07, pp. 1–8, Morristown, NJ, USA. Association for Computational Linguistics.

Barabasi, A. L.; Jeong, H.; Neda, Z.; Ravasz, E.; Schubert, A.; Vicsek, T. 2001. Evolution of the social network of scientific collaborations. *Physica A: Statistical Mechanics and its Applications*, (3-4):590 – 614.

Bejcek, Eduard; Stranák, Pavel. 2010. Annotation of multiword expressions in the Prague Dependency Treebank. *Language Resources and Evaluation*, 44(1-2):7–21.

Bohnet, Bernd. 2010. Top accuracy and fast dependency parsing is not a contradiction. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pp. 89–97, Beijing, China, August. Coling 2010 Organizing Committee.

Bonin, Francesca; Dell'Orletta, Felice; Venturi, Giulia; Montemagni, Simonetta. 2010. Contrastive filtering of domain-specific multi-word terms from different types of corpora. In *Proceedings of the Workshop on Multiword Expressions: from Theory to Applications (MWE 2010)*, pp. 76–79, Beijing, China, August. Association for Computational Linguistics.

Bouma, Gerlof. 2010. Collocation extraction beyond the independence assumption. In *Proceedings of the ACL 2010 Conference Short Papers*, pp. 109–114, Uppsala, Sweden, July. Association for Computational Linguistics.

Calzolari, Nicoletta; Fillmore, Charles; Grishman, Ralph; Ide, Nancy; Lenci, Alessandro; MacLeod, Catherine; Zampolli, Antonio. 2002. Towards best practice for multiword expressions in computational lexicons. In *Proceedings of the 3rd International Conference on Language Resources and Evaluation (LREC-2002)*, pp. 1934–1940, Las Palmas.

Caseli, Helena de Medeiros; Villavicencio, Aline; Machado, André; Finatto, Maria José. 2009. Statistically-driven alignment-based multiword expression identification for technical domains. In *Proceedings of the Workshop on Multiword Expressions: Identification, Interpretation, Disambiguation and Applications*, pp. 1–8, Singapore, August. Association for Computational Linguistics.

Caseli, Helena de Medeiros; Ramisch, Carlos; Nunes, Maria das Graças Volpe; Villavicencio, Aline. 2010. Alignment-based extraction of multiword expressions. *Language Resources and Evaluation*, 44(1-2):59–77.

Chinchor, Nancy. 1998. MUC-7 Named Entity Task Definition. In *Proceedings of Seventh Message Understanding Conference*.

Cinková, Silvie; Kolářová, Veronika. 2005. Nouns as Components of Support Verb Constructions in the Prague Dependency Treebank. In Simková, Mária (ed.), *Insight into Slovak and Cech Corpus Linguistics*, pp. 113–139. Veda Bratislava, Slovakia.

Cook, Paul; Fazly, Afsaneh; Stevenson, Suzanne. 2007. Pulling their weight: exploiting syntactic forms for the automatic identification of idiomatic expressions in context. In *Proceedings of the Workshop on a Broader Perspective on Multiword Expressions*, MWE '07, pp. 41–48, Morristown, NJ, USA. Association for Computational Linguistics.

Cook, Paul; Fazly, Afsaneh; Stevenson, Suzanne. 2008. The VNC-Tokens Dataset. In *Proceedings of the LREC Workshop Towards a Shared Task for Multiword Expressions (MWE 2008)*, pp. 19–22, Marrakech, Morocco, June.

Cortes, Corinna; Vapnik, Vladimir. 1995. Support-vector networks. *Machine Learning*, 20(3):273–297.

Csendes, Dóra; Csirik, János; Gyimóthy, Tibor; Kocsor, András. 2005. The Szeged TreeBank. In Matousek, Václav; Mautner, Pavel; Pavelka, Tomás (eds.), *Proceedings of the 8th International Conference on Text, Speech and Dialogue, TSD 2005*, Lecture Notes in Computer Science, pp. 123–132, Berlin / Heidelberg, September. Springer.

Daumé III, Hal. 2007. Frustratingly easy domain adaptation. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pp. 256–263, Prague, Czech Republic, June. Association for Computational Linguistics.

Diab, Mona; Bhutada, Pravin. 2009. Verb Noun Construction MWE Token Classification. In *Proceedings of the Workshop on Multiword Expressions: Identification, Interpretation, Disambiguation and Applications*, pp. 17–22, Singapore, August. Association for Computational Linguistics.

Dias, Gaël. 2003. Multiword unit hybrid extraction. In *Proceedings of the ACL 2003 workshop on Multiword expressions: analysis, acquisition and treatment – Volume 18*, pp. 41–48, Morristown, NJ, USA. Association for Computational Linguistics.

Doddington, George; Mitchell, Alexis; Przybocki, Mark; Ramshaw, Lance; Strassel, Stephanie; Weischedel, Ralph. 2004. The Automatic Content Extraction (ACE) Program – Tasks, Data and Evaluation. In *Proceedings of LREC 2004*, pp. 837–840.

É. Kiss, Katalin. 2002. *The Syntax of Hungarian*. Cambridge University Press, Cambridge.

Erk, Katrin; Strapparava, Carlo (eds.). 2010. *Proceedings of the 5th International Workshop on Semantic Evaluation*. Association for Computational Linguistics, Uppsala, Sweden, July.

Evert, Stefan. 2008. A lexicographic evaluation of German adjective-noun collocations. In *Proceedings of the LREC 2008 Workshop Towards a Shared Task for Multiword Expressions (MWE 2008)*, pp. 3–6, Marrakech, Morocco.

Farkas, Richárd; Szarvas, György; Kocsor, András. 2006. Named entity recognition for Hungarian using various machine learning algorithms. *Acta Cybernetica*, 17(3):633–646, January.

Farkas, Richárd; Ormándi, Róbert; Jelasity, Márk; Csirik, János. 2008. A manually annotated html corpus for a novel scientific trend analysis. In *The Eighth IAPR International Workshop on Document Analysis Systems, Nara (DAS2008)*. Extended abstracts.

Fazly, Afsaneh; Stevenson, Suzanne. 2007. Distinguishing Subtypes of Multiword Expressions Using Linguistically-Motivated Statistical Measures. In *Proceedings of the Workshop on A Broader Perspective on Multiword Expressions*, pp. 9–16, Prague, Czech Republic, June. Association for Computational Linguistics.

Fellbaum, Christiane (ed.). 1998. *WordNet An Electronic Lexical Database*. The MIT Press, Cambridge, MA ; London, May.

Finin, T.W. 1980. The semantic interpretation of nominal compounds. In *Proc. of the 1st Conference on Artificial Intelligence (AAAI-80)*.

Finkel, Jenny Rose; Grenager, Trond; Manning, Christopher. 2005. Incorporating non-local information into information extraction systems by Gibbs sampling. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, pp. 363–370, Stroudsburg, PA, USA. Association for Computational Linguistics.

Goodrum, A. A; McCain, K. W; Lawrence, S.; Giles, C. L. 2001. Scholarly publishing in the Internet age: a citation analysis of computer science literature. *Information Processing and Management*, 37:661–675, September.

Grégoire, Nicole. 2007. Design and Implementation of a Lexicon of Dutch Multiword Expressions. In *Proceedings of the Workshop on A Broader Perspective on Multiword Expressions*, pp. 17–24, Prague, Czech Republic, June. Association for Computational Linguistics.

Grégoire, Nicole. 2010. DuELME: a Dutch electronic lexicon of multiword expressions. *Language Resources and Evaluation*, 44(1-2):23–39.

Grishman, Ralph; Sundheim, Beth. 1995. Design of the MUC-6 evaluation. In *Proceedings of MUC-6*, pp. 1–12, Stroudsburg, PA, USA. ACL.

Gurrutxaga, Antton; Alegria, Iñaki. 2011. Automatic Extraction of NV Expressions in Basque: Basic Issues on Cooccurrence Techniques. In *Proceedings of the Workshop on Multiword Expressions: from Parsing and Generation to the Real World*, pp. 2–7, Portland, Oregon, USA, June. Association for Computational Linguistics.

Halácsy, Péter; Kornai, András; Németh, László; Sass, Bálint; Varga, Dániel; Váradi, Tamás; Vonyó, Attila. 2005. A hunglish korpusz és szótár [The hunglish corpus and dictionary]. In Alexin, Zoltán; Csendes, Dóra (eds.), *MSzNy 2005 – III. Magyar Számítógépes Nyelvészeti Konferencia*, pp. 134–142, Szeged, Hungary, December. University of Szeged.

Hall, Mark; Frank, Eibe; Holmes, Geoffrey; Pfahringer, Bernhard; Reutemann, Peter; Witten, Ian H. 2009. The WEKA data mining software: an update. *SIGKDD Explorations*, 11(1):10–18.

Hedlund, Turid. 2002. Compounds in dictionary-based cross-language information retrieval. *Information Research*, 7(2).

Hendrickx, Iris; Mendes, Amália; Pereira, Sílvia; Gonçalves, Anabela; Duarte, Inês. 2010. Complex Predicates Annotation in a Corpus of Portuguese. In *Proceedings of the Fourth Linguistic Annotation Workshop*, pp. 100–108, Uppsala, Sweden, July. Association for Computational Linguistics.

Jackendoff, Ray. 1997. *The Architecture of the Language Faculty*. MIT Press, Cambridge, MA.

Kaalep, Heiki-Jaan; Muischnek, Kadri. 2006. Multi-Word Verbs in a Flective Language: The Case of Estonian. In *Proceedings of the EACL Workshop on Multi-Word Expressions in a Multilingual Contexts*, pp. 57–64, Trento, Italy, April. Association for Computational Linguistics.

Kaalep, Heiki-Jaan; Muischnek, Kadri. 2008. Multi-Word Verbs of Estonian: a Database and a Corpus. In *Proceedings of the LREC Workshop Towards a Shared Task for Multiword Expressions (MWE 2008)*, pp. 23–26, Marrakech, Morocco, June.

Kearns, Kate. 2002. *Light verbs in English*. Manuscript.

Kim, Su Nam; Baldwin, Timothy. 2006. Interpreting semantic relations in noun compounds via verb semantics. In *Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions*, pp. 491–498, Sydney, Australia, July. Association for Computational Linguistics.

Kim, Su Nam; Baldwin, Timothy. 2008. An unsupervised approach to interpreting noun compounds. In *Proceedings of the 2008 IEEE International Conference on Natural Language Processing and Knowledge Engineering (NLP-KE-08)*, Beijing, China.

Kim, Su Nam; Nakov, Preslav. 2011. Large-scale noun compound interpretation using bootstrapping and the web as a corpus. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, pp. 648–658, Stroudsburg, PA, USA. Association for Computational Linguistics.

Kim, Su Nam. 2008. *Statistical Modeling of Multiword Expressions*. Ph.D. thesis, University of Melbourne, Melbourne.

Klein, Dan; Manning, Christopher D. 2003. Accurate unlexicalized parsing. In *Annual Meeting of the ACL*, volume 41, pp. 423–430.

Krenn, Brigitte. 2008. Description of Evaluation Resource – German PP-verb data. In *Proceedings of the LREC Workshop Towards a Shared Task for Multiword Expressions (MWE 2008)*, pp. 7–10, Marrakech, Morocco, June.

Lafferty, John; McCallum, Andrew; Pereira, Fernando. 2001a. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of ICML-01, 18th Int. Conf. on Machine Learning*, pp. 282–289. Morgan Kaufmann.

Lafferty, John D.; McCallum, Andrew; Pereira, Fernando C. N. 2001b. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, pp. 282–289, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Laporte, Eric; Voyatzi, Stavroula. 2008. An Electronic Dictionary of French Multiword Adverbs. In *Proceedings of the LREC 2008 Workshop Towards a Shared Task for Multiword Expressions (MWE 2008)*, pp. 31–34, Marrakech, Morocco.

Laporte, Eric; Nakamura, Takuya; Voyatzi, Stavroula. 2008. A French Corpus Annotated for Multiword Nouns. In *Proceedings of the LREC 2008 Workshop Towards a Shared Task for Multiword Expressions (MWE 2008)*, pp. 27–30, Marrakech, Morocco.

Levi, Judith. 1978. The syntax and semantics of complex nominals. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Academic Press.

McCallum, Andrew Kachites. 2002. MALLET: A Machine Learning for Language Toolkit. http://mallet.cs.umass.edu.

Meyers, Adam; Reeves, Ruth; Macleod, Catherine; Szekely, Rachel; Zielinska, Veronika; Young, Brian; Grishman, Ralph. 2004. The NomBank Project: An Interim Report. In Meyers, Adam (ed.), *HLT-NAACL 2004 Workshop: Frontiers in Corpus Annotation*, pp. 24–31, Boston, Massachusetts, USA, May 2 - May 7. Association for Computational Linguistics.

Mihaltz, Márton; Hatvani, Csaba; Kuti, Judit; Szarvas, György; Csirik, János; Prószéky, Gábor; Váradi, Tamás. 2008. Methods and Results of the Hungarian WordNet Project. In Tanács, Attila; Csendes, Dóra; Vincze, Veronika; Fellbaum, Christiane; Vossen, Piek (eds.), *Proceedings of the Fourth Global WordNet Conference (GWC 2008)*, pp. 311–320, Szeged. University of Szeged.

Muischnek, Kadri; Kaalep, Heiki Jaan. 2010. The variability of multi-word verbal expressions in Estonian. *Language Resources and Evaluation*, 44(1-2):115–135.

Nagy, István; Vincze, Veronika. 2013. English Nominal Compound Detection with Wikipedia-Based Methods. In Matousek, Václav; Mautner, Pavel; Pavelka, Tomás (eds.), *Proceedings of the 16th International Conference on Text, Speech and Dialogue, TSD 2013*, Lecture Notes in Computer Science, pp. 225–232. Springer, Berlin / Heidelberg, September.

Nagy, István; Farkas, Richárd; Jelasity, Márk. 2009. Researcher affiliation extraction from homepages. In *Proceedings of the 2009 Workshop on Text and Citation Analysis for Scholarly Digital Libraries*, NLPIR4DL '09, pp. 1–9, Stroudsburg, PA, USA. Association for Computational Linguistics.

Nagy T., István; Berend, Gábor; Vincze, Veronika. 2011a. Noun compound and named entity recognition and their usability in keyphrase extraction. In *Proceedings of RANLP 2011*, Hissar, Bulgaria.

Nagy T., István; Vincze, Veronika; Berend, Gábor. 2011b. Domain-Dependent Identification of Multiword Expressions. In *Proceedings of the International Conference Recent Advances in Natural Language Processing 2011*, pp. 622–627, Hissar, Bulgaria, September. RANLP 2011 Organising Committee.

Nagy T., István; Vincze, Veronika; Farkas, Richárd. 2013. Full-coverage Identification of English Light Verb Constructions. In *Proceedings of the Sixth International Joint Conference on Natural Language Processing*, pp. 329–337, Nagoya, Japan, October. Asian Federation of Natural Language Processing.

Nagy T., István. 2009. Összetett rendszer vállalkozások címeinek webről történő automatikus összegyűjtésére [Complex system for automatic detection of addresses of companies from Web]. In *XXIX. Országos Tudományos Diákköri Konferencia OTDK Informatikai szekció*. Debrecen.

Nagy T., István. 2012. Person attribute extraction from the textual parts of web pages. *Acta Cybernetica*, 20(3):419–440.

Nemeskey, Dávid Márk; Simon, Eszter. 2012. Automatically Generated NE Tagged Corpora for English and Hungarian. In *Proceedings of the 4th Named Entity Workshop*, NEWS '12, pp. 38–46, Stroudsburg, PA, USA. Association for Computational Linguistics.

Newman, M. E. J. 2001. The structure of scientific collaboration networks. In *Proceedings of the National Academy of Sciences of the United States of America*, volume 98, pp. 404–409, Santa Fe Institute, January.

Nicholson, Jeremy; Baldwin, Timothy. 2006. Interpretation of compound nominalisations using corpus and web statistics. In *Proceedings of the Workshop on Multiword Expressions: Identifying and Exploiting Underlying Properties*, MWE '06, pp. 54–61, Stroudsburg, PA, USA. Association for Computational Linguistics.

Nicholson, Jeremy; Baldwin, Timothy. 2008. Interpreting Compound Nominalisations. In *LREC 2008 Workshop: Towards a Shared Task for Multiword Expressions (MWE 2008)*, pp. 43–45, Marrakech, Morocco.

Nunberg, Geoffrey; Sag, Ivan A.; Wasow, Thomas. 1994. Idioms. *Language*, 70:491–538.

Pecina, Pavel. 2010. Lexical association measures and collocation extraction. *Language Resources and Evaluation*, 44(1-2):137–158.

Piao, Scott S. L.; Rayson, Paul; Archer, Dawn; Wilson, Andrew; McEnery, Tony. 2003. Extracting multiword expressions with a semantic tagger. In *Proceedings of the ACL 2003 workshop on Multiword expressions: analysis, acquisition and treatment – Volume 18*, pp. 49–56, Morristown, NJ, USA. Association for Computational Linguistics.

Quinlan, Ross. 1993. *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers, San Mateo, CA.

Ramisch, Carlos; Villavicencio, Aline; Boitet, Christian. 2010a. Multiword Expressions in the wild? The mwetoolkit comes in handy. In *Coling 2010: Demonstrations*, pp. 57–60, Beijing, China, August. Coling 2010 Organizing Committee.

Ramisch, Carlos; Villavicencio, Aline; Boitet, Christian. 2010b. mwetoolkit: a framework for multiword expression identification. In Calzolari, Nicoletta; Choukri, Khalid; Maegaard, Bente; Mariani, Joseph; Odjik, Jan; Piperidis, Stelios; Tapias, Daniel (eds.), *Proceedings of the Seventh Conference on International Language Resources and Evaluation (LREC'10)*, Valletta, Malta, May. European Language Resources Association (ELRA).

Ramisch, Carlos; Villavicencio, Aline; Boitet, Christian. 2010c. Web-based and combined language models: a case study on noun compound identification. In *Coling 2010: Posters*, pp. 1041–1049, Beijing, China, August. Coling 2010 Organizing Committee.

Ratinov, Lev; Roth, Dan. 2009. Design challenges and misconceptions in named entity recognition. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning*, CoNLL '09, pp. 147–155, Stroudsburg, PA, USA. Association for Computational Linguistics.

Sag, Ivan A.; Baldwin, Timothy; Bond, Francis; Copestake, Ann; Flickinger, Dan. 2002. Multiword Expressions: A Pain in the Neck for NLP. In *Proceedings of the 3rd International Conference on Intelligent Text Processing and Computational Linguistics (CICLing-2002*, pp. 1–15, Mexico City, Mexico.

Samardžić, Tanja; Merlo, Paola. 2010. Cross-lingual variation of light verb constructions: Using parallel corpora and automatic alignment for linguistic research. In *Proceedings of the 2010 Workshop on NLP and Linguistics: Finding the Common Ground*, pp. 52–60, Uppsala, Sweden, July. Association for Computational Linguistics.

Sanches Duran, Magali; Ramisch, Carlos; Aluísio, Sandra Maria; Villavicencio, Aline. 2011. Identifying and Analyzing Brazilian Portuguese Complex Predicates. In *Proceedings of the Workshop on Multiword Expressions: from Parsing and Generation to the Real World*, pp. 74–82, Portland, Oregon, USA, June. Association for Computational Linguistics.

Sang, Erik F. Tjong Kim; Meulder, Fien De. 2003. Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition. In *Proceedings of CoNLL-2003*, pp. 142–147.

Sanromán Vilas, Begoña. 2009. Towards a semantically oriented selection of the values of Oper_1. The case of *golpe* 'blow' in Spanish. In Beck, David; Gerdes, Kim; Milićević, Jasmina; Polguère, Alain (eds.), *Proceedings of the Fourth International Conference on Meaning-Text Theory – MTT'09*, pp. 327–337, Montreal, Canada. Université de Montréal.

Sass, Bálint. 2010. Párhuzamos igei szerkezetek közvetlen kinyerése párhuzamos korpuszból [Extracting parallel multiword verbs from parallel corpora]. In Tanács, Attila; Vincze, Veronika (eds.), *VII. Magyar Számítógépes Nyelvészeti Konferencia*, pp. 102–110, Szeged. Szegedi Tudományegyetem.

Sass, Bálint. 2013. *Igei szerkezetek gyakorisági szótára: Egy automatikus lexikai kinyerő eljárás és alkalmazása [Dictionary of verbal constructions: An automatic lexical extraction method and its application]*. Ph.D. thesis, Pázmány Péter Katolikus Egyetem, Budapest, Hungary.

Simon, Eszter. 2013. *Approaches to Hungarian Named Entity Recognition*. Ph.D. thesis, Budapest University of Technology and Economics, Budapest, Hungary.

Sinha, Rai Mahesh. 2011. Stepwise Mining of Multi-Word Expressions in Hindi. In *Proceedings of the Workshop on Multiword Expressions: from Parsing and Generation to the Real World*, pp. 110–115, Portland, Oregon, USA, June. Association for Computational Linguistics.

Spink, Amanda; Jansen, Bernard; Pedersen, Jan. 2004. Searching for people on web search engines. *Journal of Documentation*, 60:266 – 278.

Steinberger, Ralf; Pouliquen, Bruno; Widiger, Anna; Ignat, Camelia; Erjavec, Tomaž; Tufiş, Dan. 2006. The JRC-Acquis: A multilingual aligned parallel corpus with 20+ languages. In *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC'2006*, pp. 2142–2147.

Stevenson, Suzanne; Fazly, Afsaneh; North, Ryan. 2004. Statistical Measures of the Semi-Productivity of Light Verb Constructions. In Tanaka, Takaaki; Villavicencio, Aline; Bond, Francis; Korhonen, Anna (eds.), *Second ACL Workshop on Multiword Expressions: Integrating Processing*, pp. 1–8, Barcelona, Spain, July. Association for Computational Linguistics.

Szarvas, György; Farkas, Richárd; Felföldi, László; Kocsor, András; Csirik, János. 2006a. A highly accurate Named Entity corpus for Hungarian. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06)*. European Language Resources Association (ELRA).

Szarvas, György; Farkas, Richárd; Kocsor, András. 2006b. A multilingual named entity recognition system using boosting and C4.5 decision tree learning algorithms. In *Proceedings of the 9th international conference on Discovery Science*, DS'06, pp. 267–278, Berlin, Heidelberg. Springer-Verlag.

Szarvas, György; Vincze, Veronika; Farkas, Richárd; Móra, György; Gurevych, Iryna. 2012. Cross-genre and cross-domain detection of semantic uncertainty. *Computational Linguistics*, 38(2):335–367, June.

Tan, Yee Fan; Kan, Min-Yen; Cui, Hang. 2006. Extending corpus-based identification of light verb constructions using a supervised learning framework. In *Proceedings of the EACL Workshop on Multi-Word Expressions in a Multilingual Contexts*, pp. 49–56, Trento, Italy, April. Association for Computational Linguistics.

Teufel, S.; Siddharthan, A.; Tidhar, D. 2006. An annotation scheme for citation function. In *Proceedings of Sigdial-06*.

Tjong Kim Sang, Erik F.; De Meulder, Fien. 2003. Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition. In *Proceedings of CoNLL-2003*, pp. 142–147. Edmonton, Canada.

Tjong Kim Sang, Erik F. 2002. Introduction to the CoNLL-2002 Shared Task: Language-Independent Named Entity Recognition. In *Proceedings of CoNLL-2002*, pp. 155–158. Taipei, Taiwan.

Tóth, Krisztina; Farkas, Richárd; Kocsor, András. 2008. Hybrid algorithm for sentence alignment of Hungarian-English parallel corpora. *Acta Cybernetica*, 18(3):463–478.

Toutanova, Kristina; Manning, Christopher D. 2000. Enriching the knowledge sources used in a maximum entropy part-of-speech tagger. In *Proceedings of EMNLP 2000*, pp. 63–70, Stroudsburg, PA, USA. Association for Computational Linguistics.

Tu, Yuancheng; Roth, Dan. 2011. Learning English Light Verb Constructions: Contextual or Statistical. In *Proceedings of the Workshop on Multiword Expressions: from Parsing and Generation to the Real World*, pp. 31–39, Portland, Oregon, USA, June. Association for Computational Linguistics.

Van de Cruys, Tim; Moirón, Begoña Villada. 2007. Semantics-based multiword expression extraction. In *Proceedings of the Workshop on a Broader Perspective on Multiword Expressions*, MWE '07, pp. 25–32, Morristown, NJ, USA. Association for Computational Linguistics.

Varga, Dániel; Simon, Eszter. 2007. Hungarian named entity recognition with a maximum entropy approach. *Acta Cybernetica*, 18(2):293–301.

Villavicencio, Aline; Kordoni, Valia; Zhang, Yi; Idiart, Marco; Ramisch, Carlos. 2007. Validation and evaluation of automatically acquired multiword expressions for grammar engineering. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pp. 1034–1043, Prague, Czech Republic, June. Association for Computational Linguistics.

Vincze, Veronika; Szauter, Dóra; Almási, Attila; Móra, György; Alexin, Zoltán; Csirik, János. 2010. Hungarian Dependency Treebank. In Calzolari, Nicoletta; Choukri, Khalid; Maegaard, Bente; Mariani, Joseph; Odijk, Jan; Piperidis, Stelios; Rosner, Mike; Tapias, Daniel (eds.), *Proceedings of the Seventh Conference on International Language Resources and Evaluation (LREC'10)*, Valletta, Malta, May. European Language Resources Association (ELRA).

Vincze, Veronika; Nagy T., István; Berend, Gábor. 2011a. Detecting Noun Compounds and Light Verb Constructions: a Contrastive Study. In *Proceedings of the Workshop on Multiword Expressions: from Parsing and Generation to the Real World*, pp. 116–121, Portland, Oregon, USA, June. ACL.

Vincze, Veronika; Nagy T., István; Berend, Gábor. 2011b. Multiword expressions and named entities in the Wiki50 corpus. In *Proceedings of RANLP 2011*, Hissar, Bulgaria.

Vincze, Veronika; Nagy T., István; Farkas, Richárd. 2013a. Identifying English and Hungarian Light Verb Constructions: A Contrastive Approach. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 255–261, Sofia, Bulgaria, August. Association for Computational Linguistics.

Vincze, Veronika; Nagy T., István; Zsibrita, János. 2013b. Learning to detect English and Hungarian light verb constructions. *ACM Trans. Speech Lang. Process.*, 10(2):6:1–6:25, June.

Vincze, Veronika. 2011. *Semi-Compositional Noun + Verb Constructions: Theoretical Questions and Computational Linguistic Analyses*. Ph.D. thesis, University of Szeged, Szeged, Hungary.

Vincze, Veronika. 2012. Light Verb Constructions in the SzegedParalellFX English–Hungarian Parallel Corpus. In Calzolari, Nicoletta; Choukri, Khalid; Declerck, Thierry; Doğan, Mehmet Uğur; Maegaard, Bente; Mariani, Joseph; Odijk, Jan; Piperidis, Stelios (eds.), *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC-2012)*, pp. 2381–2388, Istanbul, Turkey, May. European Language Resources Association (ELRA). ACL Anthology Identifier: L12-1041.

Watanabe, Keigo; Bollegala, Danushka; Matsuo, Yutaka; Ishizuka, Mitsuru. 2009. A two-step approach to extracting attributes for people on the web. In *2nd Web People Search Evaluation Workshop (WePS 2009), 18th WWW Conference*.

Yu, Hwanjo; Han, Jiawei; Chang, Kevin Chen-Chuan. 2002. PEBL: positive example-based learning for Web page classification using SVM. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '02, pp. 239–248, New York, NY, USA. ACM.

Zsibrita, János; Vincze, Veronika; Farkas, Richárd. 2013. magyarlanc: A Toolkit for Morphological and Dependency Parsing of Hungarian. In *Proceedings of RANLP-2013*, pp. 763–771, Hissar, Bulgaria.