FASSBL7

THE SEVENTH INTERNATIONAL CONFERENCE
FORMAL APPROACHES TO SOUTH SLAVIC AND BALKAN LANGUAGES
4-6 OCTOBER 2010, DUBROVNIK, CROATIA

THE SEVENTH INTERNATIONAL CONFERENCE
FORMAL APPROACHES TO SOUTH SLAVIC AND BALKAN LANGUAGES
4-6 OCTOBER 2010, DUBROVNIK, CROATIA

Organized by

Institute of Linguistics, Faculty of Humanities and Social Sciences,
University of Zagreb

Croatian Language Technologies Society

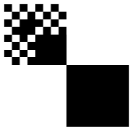
The Department of Computational Linguistics, Institute of Bulgarian Language
"Prof. Lyubomir Andreychin", Bulgarian Academy of Sciences

The Norwegian University of Science and Technology

Supported by

Ministry of Science, Education and Sports
of the Republic of Croatia

Bulgarian Academy of Sciences



Croatian Language Technologies Society

A CIP catalogue record for this book is available at the National and University Library in Zagreb under No 744527

ISBN 978-953-55375-2-6

THE SEVENTH INTERNATIONAL CONFERENCE
FORMAL APPROACHES TO SOUTH SLAVIC AND BALKAN LANGUAGES
4-6 OCTOBER 2010, DUBROVNIK, CROATIA

PROCEEDINGS

Edited by
Marko Tadić, Mila Dimitrova-Vulchanova, Svetla Koeva

Croatian Language Technologies Society – Faculty of Humanities and Social Sciences
Zagreb, 2010

Organizing committee

Damir Boras (University of Zagreb)
Svetla Koeva (The Bulgarian Academy of Sciences)
Marko Tadić – chair (University of Zagreb / Croatian Language Technologies Society)
Mila Vulchanova (The Norwegian University of Science and Technology)
Valentin Vulchanov (The Norwegian University of Science and Technology)

Programme committee

Damir Boras (University of Zagreb)
Anna Cardinaletti (Cá Foscari University, Venice)
Dan Cristea (University of Iaşi)
Damir Ćavar (University of Zadar)
Tomaž Erjavec (Institute Jozef Stefan, Ljubljana)
Giuliana Giusti (Cá Foscari University, Venice)
Svetla Koeva (The Bulgarian Academy of Sciences)
Iliana Krapova (Venice University)
Milan Mihaljević (Old Church Slavonic Institute, Zagreb)
Stelios Piperidis (Institute for Language and Speech Processing, Athens)
Vassil Raynov (The Bulgarian Academy of Sciences)
Kiril Ribarov (Charles University, Prague)
Melita Stavrou (Aristoteles University, Thessaloniki)
Marko Tadić (University of Zagreb / Croatian Language Technologies Society)
Dan Tufiş (The Romanian Academy)
Duško Vitas (University of Belgrade)
Mila Vulchanova – chair (The Norwegian University of Science and Technology)
Valentin Vulchanov (The Norwegian University of Science and Technology)
Chris Wilder (The Norwegian University of Science and Technology)

TABLE OF CONTENTS

ORAL PRESENTATIONS

Ileana Comorovski
Licensing Indefinite Subjects in Romanian Constituent Questions 7

Rositsa Dekova, Petya Nestorova
Formal Description of Some Intransitive Verbs of Non-Directed Movement in the Bulgarian Framenet 13

Georgi Iliev
Towards Rule-Based Optimization of Machine Translation: Applying Predicate Logic to Generate Noun-Phrase Translations from Swedish to Bulgarian 19

Radu Ion, Dan Tufis, Tiberiu Boros, Alexandru Ceausu, Dan Stefanescu
On-Line Compilation of Comparable Corpora and Their Evaluation 29

Svetla Koeva
Syntactic Annotation in Bulgarian National Corpus 35

Svetla Koeva, Svetlozara Leseva, Ekaterina Tarpomanova, Borislav Rizov, Tsvetana Dimitrova, Hristina Kukova
Bulgarian Sense-Annotated Corpus – Results And Achievements 41

Vedrana Mihalicek
(Not So) Free Word Order in Lambda Grammar: The Case of BCMS 49

Jure Mijić, Bojana Dalbelo Bašić, Jan Šnajder
Robust Keyphrase Extraction For A Large-Scale Croatian News Production System 59

Mladen Mikša, Jan Šnajder, Bojana Dalbelo Bašić
Correcting Word Merge Errors in Croatian Texts 67

Pinar Öztürk, Mila Vulchanova, Christian Tumyr, Liliana Martinez, David Kabath
Assessing the Feature-Driven Nature of Similarity-Based Sorting of Verbs 77

Teodora Radeva-Bork
Clitic Doubling in Bulgarian: Between Optionality and Obligatoriness 89

Stanimir Rakić
On the Trochaic Feet, Extrametricality and Shortening Rules in Standard Serbian 97

Ivelina Stoyanova
Factors Influencing the Performance of Some Methods for Automatic Identification of Multiword Expressions in Bulgarian 103

Jan Šnajder, Bojana Dalbelo Bašić
A Computational Model of Croatian Derivational Morphology 109

Krešimir Šojat, Željko Agić, Marko Tadić
Verb Valency Frame Extraction Using Morphological And Syntactic Features Of Croatian 119

Radka Vlahova, Atanas Atanasov
Prepositional Phrases In Bulgarian 127

Kristina Vučković, Željko Agić, Marko Tadić
Sentence Classification and Clause Detection for Croatian 131

LICENSING INDEFINITE SUBJECTS IN ROMANIAN CONSTITUENT QUESTIONS

Ileana Comorovski

Département de Sciences du langage, Université Nancy 2
23, Bd. Albert 1er, 54015 Nancy, France
ileana.comorovski@univ-nancy2.fr

1. Data, issues, and main proposal

This paper considers non-generic constituent questions with an indefinite subject. We will argue that the subject of a constituent question can be indefinite under almost the same conditions which allow the subject of a declarative to be indefinite.

To explain the distribution of indefinite subjects in certain copular constituent questions, we will examine some aspects of the information structure of questions. We will propose that sentences can have as a topic an indefinite noun phrase provided the noun phrase denotes an entity presented from a subjective point of view. The role of subjectivity is illustrated by the contrast in acceptability between (1a) and (1b) below:

(1) a. Care ar fi (după tine) *un* hotel ***(confortabil / bun)** la Sibiu?
'What would be (according to you) a comfortable/good hotel in Sibiu?'

b * Care ar fi *un* hotel **de trei stele** la Sibiu?
what would be a hotel of three stars in Sibiu

In sentence (1a), the noun *hotel* is modified by the subjective adjective *confortabil* ('comfortable'), whereas in (1b) no subjective modifier occurs. In what follows, we will take a closer look at the role of subjectivity in the licensing of indefinite subjects.

2. Indefinite subjects and topichood in constituent questions

Just like declarative sentences, constituent questions need an 'aboutness' topic: they are asked *about* something (cf. Mathesius 1915, Reinhart 1981, a.o. on the topic of declaratives, and Krifka 2001, Dikkers 2004 a.o. on the topic of interrogatives). In the words of Dikkers 2004, the topic of a question is 'the thing someone intends to increase his/her knowledge about by using the question'.

In general, topical noun phrases denote discourse-old entities (Prague school, Portner and Yabushita 1998, a.o.). Specific indefinites can also function as topics, as shown by Cresti 1995. We take specific indefinites to refer to individuals that the speaker has in mind. Unlike declaratives, questions do not allow a specific indefinite as a topic. The explanation for this fact is straightforward: if the topic were a specific indefinite, the question would be used to request information about an entity known to the speaker, but unknown to the hearer. The hearer could therefore not answer the question and the very purpose of the interrogative speech act would be defeated.

If the word order in a sentence is neutral and the subject is definite, the sentence is usually about the subject. If the subject is indefinite, the sentence can be about: a) a non-subject constituent; b) an (implicit) event argument or, as we will argue, c) it can have as a topic the indefinite subject itself, provided the subject denotes an entity presented from a subjective point of view.

Below are examples of the three types of topics as they occur in constituent questions:

a) Questions that are about an argument projected as a constituent other than the subject. This constituent is often fronted sentence-initially:

(2) Noul roman al Iuliei, oare când îl va publica *cineva* / *(vre)o editură*?
'Julie's new novel, when will *someone/some publishing house* publish it?'

The topic of (2) is the fronted DP. Following Krifka (2001:35), we allow topics to scope out of speech acts. For instance, the topic of (2) scopes out of a question act, as represented below:

(2') *Top* [Julie's new novel] λx_1[*Quest* [when will somebody publish it_1]

In sentence (2), the resumptive pronoun *it* is associated with the topic constituent *Julie's new novel*. The string *when will somebody publish it* is interpreted as a complex predicate; the string contains the linguistic expression of a variable, namely the resumptive pronoun *it*. The complex predicate is predicated of the topic *Julie's new novel*. In the representation (2'), the λ-operator binds the resumptive pronoun; we have coindexed *x* and *it* so as to represent the variable binding relation.

b) Questions that are about an (implicit) event argument:

(3) Când va mai trece *cineva / (vre)un autobuz* pe aici?
'When will someone / a bus pass by here again?

The answer to question (3) is a thetic statement, i.e. a statement analyzed by the Prague school as not being divided into a topic part and a comment part ('unpartitioned' or 'all new information' sentence). Jäger 2001 proposes that thetic statements are about an (implicit) event that is an argument of the verb (cf. Davidson's 1967 event argument). In the same vein, we suggest that the topic of question (3) is an implicit argument, namely someone's passing by this place, an event which is presupposed by the constituent question and is therefore a discourse-old entity.

c) Questions that are about an entity presented from a subjective point of view, as in (1a) above; the type of topic of (1a) will be presented in section 3.

3. Topichood, frame setting, and point of view

To illustrate the role of subjectivity with respect to topichood, we will work with a particular type of sentences, namely specificational copular sentences, first studied by Higgins 1973. These are sentences of the type *DP-copula-DP*. We have chosen sentences of this type because their very simple syntax and argument structure offers few candidates for topichood and thus facilitates the teasing apart of the role subjectivity plays with respect to information structure.

Here is Higgins's 1973 classification of copular sentences:

(4)	a. predicative	Tom is a novelist / brave.
	b. identity (equative)	The Morning Star is the Evening Star.
		Ion Barbu is Dan Barbilian.
	c. specificational	The winner of the election is Joe Smith.
		The guests are Jane and Tom.
	d. identificational	That is Jane.
		That woman is Jane.

Higgins informally characterizes specificational copular sentences as having a subject that acts as the heading of a list, with the complement of the copula exhaustively enumerating the members of the list.

Interestingly, when considering definite subjects, Higgins emphasizes the fact that in specificational clauses the subject is not 'referential', but has an attributive-like reading. Comorovski (2007) and Romero (2005) argue on independent grounds that the subject of a specificational clause is intensional. Comorovski demonstrates that the subject cannot be a rigid designator, but is of type <s,e>, i.e. it denotes an individual concept (=a functions from indices to individuals). For instance, in the second sentence in (4c), *The winner of the election is Joe Smith*, the subject denotes a function from indices (=world-time pairs) to the winner of the election at every index. The complement of the copula, *Joe Smith*, is the value of the function at the actual index.

In case the subject is indefinite, it can be of type <s,e> only if it is not specific, since, as pointed out by Yeom 1998, a specific indefinite functions as a rigid designator. Of course, rigidity is restricted here to the belief (or information state) of the agent who 'has an individual in mind'.

Specificational sentences have been recently analyzed as having the subject DP as their topic (Geist 2007, Mikkelsen 2004, a.o.). This generalization is not without problems, as the subject can be indefinite (e.g. *One person who might help you is Mary* (Higgins 1973: 270)). We refer the reader to Chapter 8 of Mikkelsen 2004 for insightful discussion of the issue of declarative specificational sentences with indefinite subjects.

Specificational sentences can also take the form of constituent questions, as argued by Comorovski 2007. In Romanian, specificational questions have the form $DP_{[+wh]}$*-copula-DP*, as illustrated by (1a) above and by (5) below:

(5) a. Care e [$_{DP}$ capitala Moldovei]?
'What is the capital of Moldavia?'

b. Care e [$_{DP}$ temperatura (la voi)]?
'What is the temperature (where you are now)?'

Comorovski 2008 adduces syntactic and semantic evidence that the subject of Romanian specificational wh-questions is the postcopular DP.

We have seen that, just like declaratives, questions have a topic. The topic of (5a) is the definite noun phrase *capitala Moldovei*. What can be the topic of (1a)? The interrogative DP *care* is ruled out, since interrogative phrases cannot serve as topics. This can be easily demonstrated by looking at languages that have a topic marker, such as Japanese. In the Japanese question (6a) below, the topic is 'John'; this is indicated by the topic-marker *wa*. Kuno (1972) observes that thematic *wa* cannot appear with an interrogative phrase; this is seen in (6b):

(6) a. John-**wa** nani-o yon-da-no (example from Tomioka 2007)
John-TOP what-Acc. read-Past-Qprt.
'What did John read?'
b. Dare *-wa / -ga kita-no
who-TOP/Acc. come-Past-Qprt.
'Who came?'

The other candidate for topichood in (1a) is the postcopular indefinite DP, which is the syntactic subject of the question. We will investigate the properties of this indefinite DP.

The indefinite DP *un hotel confortabil* contains the subjective adjective *confortabil*. The hotel is comfortable from somebody's point of view. The point of view is held by what Lasersohn 2005 calls a 'judge'. Lasersohn treats the judge as one of the parameters with respect to which sentences that contain subjective predicates are evaluated.

The point of view of a judge can function as the *frame setter* of a sentence, a term we use in the sense of Jacobs 2001. Jacobs (2001: 656) defines a frame setter as follows: "In (X Y), X is the frame setter for Y iff X specifies a domain of (possible) reality to which the proposition expressed by Y is restricted."

Note that the phrase that expresses the holder of the point of view can occur sentence-initially, a position typical of frame-setters:

(7) **După tine / Ion**, care ar fi un hotel bun la Sibiu?
'According to you/John, what would be a good hotel in Sibiu?'

The sentence-initial position can be taken as an overt indication of the fact that the sentence is under the scope of the point-of-view frame setter. Point-of-view frame setters switch the perspective of the previous discourse: from an objective perspective to a subjective one or from a subjective perspective to another subjective one. We suggest that it is this break produced by a point of view frame setter which makes it possible for the topic of sentences like (1b) and (7) to be indefinite and carry new information.

Thus the acceptable (1a) has a topic, as any sentence must, whereas (1b), which contains no subjective predicate, is topicless and thereby unacceptable. Note that the topic of (1b) cannot be an implicit event, since the verb in this sentence is stative and therefore does not have an event argument.

Since in our analysis the topic of (1a) is the subject, specificational wh-question with an indefinite subject are not a counterexample to Geist's 2007 and Mikkelsen's 2004 generalization that the topic of a specificational sentence is the subject.

As different from (1b), the acceptable (1a) is evaluated with respect to an index that comprises not only a world and a time <w,t>, but also a judge, *j*.[1] It is the judge parameter that allows the indefinite subject to function as a topic. Our analysis is represented below:

(8) [[*Top* [a good hotel in Sibiu] λx[*Quest* [what would be x]]]^{M,j,w,t}

In contrast to (1a), (1b) is evaluated with respect to a <w,t> index, not a <j,w,t> index. The absence of the judge from the index of evaluation blocks the possibility of having an indefinite as a topic. Since the only available candidate for topichood in (1b) is the indefinite subject, (1b) has no topic, the question is not 'about' something, and is therefore ruled out on semantic/pragmatic grounds.

4. Evidentiality

In (1a), the conditional mood of the copula is preferred over the indicative. The same can be observed in other specificational questions that contain a subjective adjective in the subject:

(9) Care ar fi un loc *frumos* de mers în vacanţă?
'What would be a beautiful place to go to for the holidays?'

We suggest that the conditional mood morphology is related to the presence of a subjective predicate in the question. Following *Gramatica limbii române* (2005: vol.I, p.376, vol.II, p.679, 688), we consider that the conditional mood can function as an evidential marker, namely as an indirect evidential. Indirect evidentials are analyzed by Izvorski 1997 as epistemic modal operators with a presupposition of available indirect evidence. We suggest that in questions such as (1a) and (9) the indirect source of evidence presupposed by the conditional mood is the judge (who can be the hearer or a third party, in case 'according to you' is replaced, for instance, by 'according to him'). The utterer of the question requests the hearer to give what he considers to be the true answer to the question from the point of view of the judge. The use of the conditional mood indicates the distancing of the utterer of the question from the objectivity of the truth of the answer.

5. Conclusion

We have explained the distribution of indefinite subjects in non-generic constituent questions in terms of information structure. We have started from the generalization that any sentence needs a topic and from the classical observation that if the subject is definite or specific, the topic of the sentence is generally the subject. We have shown that a constituent question can have an indefinite subject if the question has an available non-subject topic: a phrase fronted sentence-initially or an (implicit) event argument of the verb. Furthermore, we have shown that a constituent question can have an indefinite subject if the subject can itself function as the topic of the question. This happens if the subject contains a subjective modifier. We have explained the relation between indefinite topics, which are carriers of new information, and subjectivity in terms of the switch in perspective that occurs when a sentence is evaluated with respect to a new value of the 'judge' parameter.

[1] We put aside the context parameter, as deixis is not relevant to the present discussion.

References

Comorovski, I. 2007. Constituent Questions and the Copula of Specification. In I. Comorovski and K. von Heusinger, eds., *Existence: Semantics and Syntax*. Dordrecht: Springer.

Comorovski, I. 2008. Intensional Subjects and Indirect Contextual Anchoring. In J. Gueron and J. Lecarme, eds., *Time and Modality*. Dordrecht: Springer.

Cresti, D. 1995. *Indefinite Topics*, Ph.D. dissertation, M.I.T.

Davidson, D. 1967. The Logical Form of Action Sentences. In *Essays on Actions and Events*. Oxford: Clarendon Press.

Dikkers, E. J. W. 2004. *Natural Answer Presentation through Revision of Syntactic Patterns*, M.A. thesis, University of Twente.

Geist, L. 2007. Predication and Equation in Copular Sentences: Russian vs. English. In I. Comorovski and K. von Heusinger, eds., *Existence: Semantics and Syntax*. Dordrecht: Springer.

Higgins , F. R. 1973. The Pseudo Cleft Construction in English, Ph.D. dissertation, M.I.T.

Jacobs, J. 2001. The Dimensions of Topic-Comment. *Linguistics* 39, 641-681.

Izvorski , R. 1997. The Present Perfect as an Epistemic Modal. In *Proceedings of SALT VII*, Cornell University: CLC Publications.

Jäger, G. 2001. Topic-Comment Structure and the Contrast between Stage-Level and Individual-Level Predicates, *Journal of Semantics* 18, 83-126.

Krifka, M. 2001. Quantifying into Question Acts, *Natural Language Semantics* 9, 1-40.

Kuno, S. 1972. Functional Sentence Perspective. A Case Study from Japanese and English, *Linguistic Inquiry* 3, 269-320.

Lasersohn, P. 2005. Context Dependence, Disagreement, and Predicates of Personal Taste. *Linguistics and Philosophy* 28, 643-686.

Mathesius, V. 1915. O passivu v moderní angličtině, *Sbornik filologický* 5, 198-220.

Mikkelsen, L. 2004. *Specifying Who: On the Structure, Meaning, and Use of Specificational Copular Clauses*, Ph.D. dissertation, University of California, Santa Cruz.

Portner, P. and K. Yabushita .1998. The Semantics and Pragmatics of Topic Phrases, *Linguistics and Philosophy* 21, 117-157.

Reinhart, T. 1981. Pragmatics and Linguistics: An Analysis of Sentence Topics, *Philosophica* 27, 53-94.

Romero, M. 2005. Concealed Questions and Specificational Subjects, *Linguistics and Philosophy* 28, 687-737.

Tomioka, S. 2007. Pragmatics of LF Intervention Effects: Japanese and Korean Wh-Interrogatives, *Journal of Pragmatics* 39, 1570-1590.

Yeom, J.-L. 1998. *A Presuppositional Analysis of Specific Indefinites: Common Grounds and Structured Information States*. New York: Garland.

FORMAL DESCRIPTION OF SOME INTRANSITIVE VERBS OF NON-DIRECTED MOVEMENT IN THE BULGARIAN FRAMENET[1]

Rositsa Dekova, Petya Nestorova[2]

Department of Computational Linguistics, Institute for Bulgarian Language, Bulgarian Academy of Sciences
52 Shipchenski prohod Blvd, build. 17, Sofia 1113, Bulgaria
rosdek@dcl.bas.bg, petyanestorova@dcl.bas.bg

ABSTRACT

The article examines the argument structures of a group of intransitive verbs of non-directed movement and their formal description in the Bulgarian FrameNet (Koeva, 2008; Koeva & Dekova, 2008). The verbs to be discussed include the Bulgarian verbs *блуждая 'roam', бродя 'rove', лутам се 'wander', скитам 'ramble,'* and *шляя се 'stroll/loaf about'*. They all share a common meaning of non-directed movement and they also display corresponding argument structures. We argue that the proper formal description of the semantic features of a verb is crucial in predicting the verb's syntactic behavior. This is particularly evident when describing a group of semantically related verbs.

1. Introduction

The Bulgarian verbs *блуждая 'roam', бродя 'rove', лутам се 'wander', скитам 'ramble,'* and *шляя се 'stroll/loaf about'* are all basic verbs, that is, they are not derived from another verb. They are also very similar in meaning – they all denote a non-directed movement which can sometimes be chaotic. Therefore, the examined verbs commonly fall under the same general definition – "to move purposelessly." Figure (1) below shows an illustrative picture of the first part of the data included in the Bulgarian FrameNet. This part contains the definition and the morpho-syntactic features of the verb *бродя 'rove,'* together with at least five example sentences[3].

	A	B	C	D	E	F	G	H	I	J
1	lemma	бродя	definition	вървя без определена посока и маршрут	from:	WN базирана	ID	"="ENG20-02043041-v	frequency	0‰
2								ENG20-01827001-v		0.21‰
3	subjectivity	transitivity	perfectiveness	inflectional type						
4	личен	непреходен	несвършен							
5										
6	1st example									
7	The example is from the text archive of DCL									
8	В същото време {по нашите планини} бродят {800 диви екземпляра}.									
9	2nd example									
10	The example is from the text archive of DCL									
11	{Децата ми} бродят {по неосвоените кътчета}.									
12	3rd example									
13	The example is from the text archive of DCL									
14	{Ако синът ми} броди гладен {из горските дебри}, ако линее в окови, ако тялото му лежи непогребано, тогава наистина ще побързам.									
15	4th example									
16	The example is from the text archive of DCL									
17	Старият Улф го открил {s} да броди {сред възвишенията на Кентъки} близо до границата с Индиана.									
18	5th example									
19	The example is from the text archive of DCL									
20	Въпреки че бе жена със светски опит, нейният свят бе застлан със зелени килими на ливадите, {в който} бродят {само крави} и шуми вятърът, а едно тихо семе									
21										

Figure 1: Formal description of the verb бродя 'rove' in the Bulgarian FrameNet (part1)

2. Argument structure

As already mentioned, the verbs under discussion share a common meaning and therefore it was expected that they also display corresponding argument structures. Indeed, the argument structures of all of the examined verbs are almost identical

[1] The financial support is granted under Contract No. BG051PO001-3.3.04/27 of 28 August 2009 within the Operation Support to the development of PhD students, post-doctoral students, post-graduate students and young scientists of the General Directorate Structural Funds and International Educational Programmes with the Ministry of Education, Youth and Science.
[2] The authors would like to thank the anonymous reviewer of the paper, whose remarks helped us to improve this article.
[3] The examples are taken either from the Bulgarian National Corpus (BulNC, available at: http://search.dcl.bas.bg/) which is collected by the people at the Department of Computational Linguistics and the Department of Bulgarian Lexicology and Lexicography at the Institute for Bulgarian Language, Bulgarian Academy of Sciences, or from other texts freely available on the Internet, such as the virtual library *Slovoto* or the newspaper *Sega*.

on both semantic and syntactic levels. Each argument structure consists of two arguments, that is, an external argument (the subject) and an internal argument (a PP with a locative function).

2.1 Syntactic-semantic specification of the external argument

The external argument can be expressed by a noun (or an NP) which denotes an animated object, a person or an animal. An interesting finding is that the first argument does not have to be the conscious doer of the action represented by the examined verbs. This refers especially to arguments expressed by nouns denoting an animal, as animals are known to act not consciously but according to their instincts, which is also evident in the corpus data (see the examples in (1) below).

(1) a) Според последното преброяване *в "Мазалат"* **бродят** 150 диви свине, 120 сърни, 100 елена и 30 мечки (BulNC);
'According to the last census 150 wild-boars, 120 doe, 100 deer, and 30 bears **rove** *in "Mazalat."*

b) Тук може да се прекарат незабравими дни и през лятото, когато планината е по-достъпна и *из нея* човек **може да скита** със седмици (BulNC);
'Unforgettable days can be spent here, when the mountain is more accessible and one can **ramble** *there* for weeks.

c) А *между тях* **се лутат** рой врани, орли, псета. (BulNC);
'A flight of crows, eagles, dogs **wander** *among them.*

d) Разбира се, сега кучетата **се шляят** *по целия остров*…(BulNC).
'Of course, now the dogs **loaf about** *on the entire island*…

It is important to mention that the corpus data provided evidence that human subjects may also be regarded as acting unconsciously, as seen in the example in (2) below.

(2) Хората **блуждаеха** *по улиците* безмълвни и унесени като зомбита. (BulNC);
'The people **roamed** *the streets* silent and rapt as zombies.'

The presence of the comparative phrase *като зомбита* (*'as zombies'*) does prove the possible lack of conscious control over the action performed by the external argument of the sentence in (2), that is *Хората* ('The people').

On the syntactic level, the subject does not have to be explicitly mentioned. This is one of the fundamental characteristics of Bulgarian as a Balkan language and therefore we do not intend to discuss it any further in this paper.

2.2 Syntactic-semantic specification of the internal argument

The actions denoted by the verbs *блуждая 'roam', бродя 'rove', лутам се 'wander', скитам 'ramble,'* and *шляя се 'stroll/loaf about'* imply also a location where the specified movement takes place. Hence, we can expect to have a syntactic component with a locative function expressed overtly whenever these verbs are used. This participant (the internal argument or the complement) is semantically related to a comparatively large area. That is, the actions denoted by the verbs at hand usually cannot be executed in a small region, as for example 2 m^2. Instead, they all entail a non-directed movement in larger space,[4] as illustrated by the phrases in italics in the examples in (1) and (2) above.

Therefore, whenever present the internal argument inevitably denotes some kind of a space, a PP with a locative function on the syntactic level. The PP is introduced by one of the following spacial prepositions: *из 'through', по 'on', в 'in', сред 'among', между 'between', край 'along', покрай 'along', около 'around', зад 'behind', пред 'in front of'*. However, only the first three prepositions are discussed presently, being the most interesting ones on the syntax-semantic interface.

It is worth mentioning that the examined verbs do not require the syntactic realization of the internal argument in the same extent. The corpus data show that the verbs *блуждая 'roam', лутам се 'wander',* and *шляя се 'stroll/loaf about'* are often

[4] The only exemption is the Bulgarian verb *лутам се 'wander'* which sometimes can be related to a smaller area. For ex. ***Лутам се*** *из стаята и не мога да си намеря място.* (*I wander around the room and I can't find a place.*)

used without the internal argument, while the verb *скитам 'ramble'* and most of all the verb *бродя 'rove'* allow for this syntactic drop out only in cases of iteration or habitual action[5].

(3) a) Струваше си да **блуждая** две хиляди години, за да науча това (BulNC)
'It was worth **roaming** for two thousand years to learn this'

b) … един слуга ми каза да не **се шляя** вечерта, … (BulNC)
'… a servant told me not to **stroll about** in the evening, …'

c) **Лутал** се безпризорен 13 дни. (BulNC)
'(He) has wandered homeless for 13 days.'

d) Хубаво беше само това, дето сега не **скиташе** много нощем. (BulNC)
'The only good thing was that now he didn't **ramble** much at nights.'

e) Най-често **бродел** неуморно пешком, като се подпирал на жезъл … (BulNC)
'Most often (he) **roved** tirelessly on foot, resting against a scepter …'

Figure (2) below shows an illustrative picture of the second part of the data included in the Bulgarian FrameNet for the verb *бродя 'rove'*. This part contains the number and the semantic specification of its arguments, described in the relevant frames[6].

	A	B	C	D	E	F	G	H	I	J
21										
22					semantic features					
23	1st frame	category	expliciteness	function	selectivity	partitiveness	semantic	preposition	complemetisers	notes
24	1st argument	NP	не	подлог	одушевено	единичен обект	entity			
25						група				
26										
27	2nd argument	PP/AdvP	да	обстоятелство	неодушевен	единичен обект	location	из, в		
28										
29										
30					semantic features					
31	2nd frame	category	expliciteness	function	selectivity	partitiveness	semantic	preposition	complemetisers	notes
32	1st argument	NP	не	подлог	одушевено	единичен обект	entity			
33						група				
34										
35	2nd argument	PP/AdvP	да	обстоятелство	неодушевен	единичен обект	location	по		surface
36										
37										

Fig. 2 *Formal description of the verb бродя 'rove' in the Bulgarian FrameNet (part2)*

Being used with this particular sense of the verbs at hand, *из 'through', по 'on', and в 'in'* have a synonymous meaning, that is, „*movement with no particular direction* within a certain area."[7] The semantics of the PPs introduced by these prepositions can be defined as *the place where the action denoted by verbs of movement permeates*. We would like to add that *place* should be taken in its broad sense and must be interpreted also as *space*, as for example *in the neighborhood*, *in our garden*, *in the woods*, etc.

In order for this meaning to be realized, a noun denoting a relatively large area must be the complement of the PP headed by these prepositions, as illustrated by the nouns in italics the above listed examples in (1a), (1b), (1d), and (2). These prepositional phrases have a locative function and they are the internal argument in the argument structure of the verbs at hand.

Although we claim that the three prepositions are synonymous in this usage, it does not mean that they can be freely interchanged at all times. The examples found in the corpora provide evidence for semantic features that distinguish the nouns that can be combined with the prepositions *из 'through'* and *в 'in'* and those, which can be combined with the preposition *по 'on.'* Most likely, it is the special relation denoted by the preposition *по 'on'* which is essentially linked to the semantics of the "surface onto which the action takes place..." (GCBL 1983:436). Thus, the preposition *в 'in'* imposes certain restrictions and they are related to nouns denoting a surface, such as *floor*, *street*, etc.

[5] This syntactic change triggered by the meaning of the verb whenever a semantic component of iteration or habitual action is present has received wide attention both in English and Bulgarian literature. Therefore, we do not feel it is necessary to discuss it in this paper.
[6] Cf. Koeva & Dekova (2008) for a detailed explanation on the formal descriton used in the Bulgarian FramenNet.
[7] However, *по 'on'* bears no association to any boundaries as the preposition *из 'through'* does. (GCBL, 1983:436)

(4) а) Създанието ... тръгна да ***се шляе*** *по / *в пода*...;
'The creature ... began to ***loaf about*** *on / *in the floor*…;

b) Хората ***блуждаеха*** *по / *в улиците.*
'People ***roamed*** *on / *in the streets.'*

Presently, we can only state that the PPs headed by *из 'through'* or *в 'in'* allow for an NP denoting space with no particular territory or borders, such as *мрак 'dusk', мъгла 'fog', тъмнина 'darkness', пустош 'wasteland', околност 'vicinity'*, etc.

3. Some predictable syntactic changes triggered by changes in verb semantics

Depending on the components present in the semantic structures of the examined verbs there are also changes in their syntactic properties. A widely discussed phenomenon on the syntax-semantic interface is a change in argument structure triggered by the semantic component usually marked as [+perfectiveness]. It is necessary to clarify that the discussed verbs as basic verbs cannot be used in the perfective voice. Instead, Bulgarian uses prefixes to form new verbs to denote perfectiveness of the situation denoted by the imperfective verb. However, some verbs, as for example *блуждая 'roam',* cannot be used to form a perfective verb. The lexemes *пребродя 'traverse / go through', избродя* '*travel all over*' are formed by prefixation from the base verb *бродя* '*rove'* which serves as the root of the newly formed verbs. However, the prefix modifies the meaning of the verb which presupposes changes in the semantic properties and the syntactic structure of the verb. The prefix *пре-* in the verb *пребродя 'traverse / go through'* adds to the meaning of the main verb (*бродя* '*rove*') the following modification: 'the action of the base verb is brought to a result as it is distributed over the whole object' (GCBL 1983:223). The verb *бродя 'rove'* does not include in its semantic structure information about any beginning, end, or result of the action, and it does not specify what part of the object (in this case the territory) is covered by the action. The prefix *из-* in the verb *избродя 'travel all over'* carries the same meaning: 'the action of the base verb is brought to a result as it is distributed over the whole object' (GCBL 1983:219). The change in the meaning of the two newly formed verbs leads to an alteration of their morphological features. The verbs are transitive and perfect in aspect. These new characteristics are related to differences in the argument structures of the derived verbs as compared to the argument structure of the base verb. Their internal argument is no longer a PP, but an NP – a direct object, which, however, retains the semantics of the internal argument of the base бродя 'rove' – the area, onto which the action is spread. Compare the underlined phrases in the example in (5a) with those in (5b) and (5c).

(5) а) Според последното преброяване *в "Мазалат"* **бродят** 150 диви свине, 120 сърни, 100 елена и 30 мечки (BulNC);
'According to the last census 150 wild-boars, 120 doe, 100 deer, and 30 bears **rove** *in "Mazalat"*.'

b) Трябваше само да **преброди** *гората*… (BulNC);
'He just had to **traverse** the woods.'

c) Човекът-шапка беше **изброди**л *всички пътеки и пътища* (BulNC);
'The hat-man has **traveled** all the paths and roads.'

The argument structure of the two new verbs differs from that of the verb *бродя 'rove'*, but the semantic structure of their arguments remains the same. The argument structure of бродя 'rove' includes an internal argument which is a PP with a locative function, while the argument structure of the verbs *пребродя 'traverse / go through'* and *избродя 'travel all over'* contain an internal argument that is an NP, which, however, also has a locative function, in parallel to the PP. That is, the semantic structure of the internal argument of all three verbs is the same and essentially means 'surface onto which the action takes place'. The only difference is its syntactic realization. There are no changes in the external argument, neither in the semantic structure, nor on the syntactic level –it is again expressed by a noun (or an NP) which denotes an animated object, a person or an animal, as illustrated by the examples in (6) below.

(6) а) В голямата гора се заселила мечка стръвница$_1$. *Много планини* t1 била **пребродила** … (BulNC);
'A meat-eating bear$_1$ has settled in the big forest.' (She / t1) has **traveled across** *many mountains* …

b) Човекът-шапка беше **изброди**л *всички пътеки и пътища* (BulNC);
'The hat-man has **traveled** *all the paths and roads*.'

The locative argument of the verb *пребродя 'traverse / go through'* is often elucidated by the words *всичко 'all'* or *цялото* '*the whole*', which essentially repeat the meaning introduced by the prefix *пре-*, as shown in the examples in (7).

(7) a) Вече си мислех, че ще трябва да **пребродя** *цялото* имение, за да те открия (BulNC);
'I've already thought that I'd have to **cross** the estate *all over* to find you .'

b) Бях готов да **пребродя** …, *всички* улици, … (BulNC);
'I was ready to **travel across** *all* streets.'

c) Като **преброди** така *цялата* мера, той отново се върна по същия път … (BulNC);
'When (he) **traveled** like this across *the whole* common land, he returned back on the same road …'

d) Дори да **пребродите** *целия* този обширен континент, … (BulNC);
'Even if you **travel** across this *whole* vast continent, ...'

The corpus data show that the locative argument in the syntactic realization of the verb *избродя 'travel all over'* also includes the semantics of *всичко 'all'* or *цялото* '*the whole*', as demonstrated in the examples in (8) below.

(8) a) … и **изброди** цялата околия *от край до край* … (BulNC);

'… and (he/she) **crossed** the whole region *from one end to another …'*

b) Сега ми се налагаше да живея в тази империалистическа държава, да я **избродя** *от край до край*, … (BulNC);
'Now I wished that I lived in this imperialistic country, to **cross** it *from one end to another …'*.'

c) Не се и съмнявам, …, че може да **избродите** *всичките мотели в страната*, … (BulNC);
'I don't even doubt it, …, that (you) can **go through** *all the motels in the country*, ….'

4. Conclusion and future perspectives

To conclude, we claim that the Bulgarian verbs *блуждая 'roam', бродя 'rove', лутам се 'wander', скитам 'ramble,'* and *шляя се 'stroll/loaf about'* are synonymous both semantically and on the syntactic level. That is, they share a common meaning and argument structure, and their arguments display similar features on the syntax-semantic interface. Using this group of closely related verbs, we show that the proper formal description of the semantic features of a verb facilitates us in predicting and describing the verb's syntactic behavior. An example was seen in the relationship of these verbs to some of their derivatives (in this case perfective verbs and their semantic and syntactic properties).

As a future perspective, we would like to introduce an elaborate comparison of the discussed verbs with verbs which can denote similar kind of aimless or even chaotic movement, whose argument structures are either parallel to those of the verbs examined so far or belong to other frames.

References

BulNC (Bulgarian National Corpus): http://search.dcl.bas.bg/

GCBL 1983. Граматика на СБКЕ. Т. II, *Морфология*, 1983, 511 p. (Grammar of Contemporary Bulgarian Language)

Koeva, S. and Dekova, R. 2008. Bulgarian FrameNet, In: Tadic et al. (eds), *Proceedings to The Sixth International Conference Formal Approaches to South Slavic and Balkan Languages*, 25-28 September 2008, Dubrovnik, Croatia, pp. 59-67.

Koeva et al. 2008. Представяне на лингвистичната информация в Българския ФреймНет – лингвистична мотивировка. In: S. Koeva (comp). *Българският ФреймНет. Семантико-синтактичен речник на българския език*. IBL-BAS, Sofia, 2008, 104 p.

TOWARDS RULE-BASED OPTIMIZATION OF MACHINE TRANSLATION: APPLYING PREDICATE LOGIC TO GENERATE NOUN-PHRASE TRANSLATIONS FROM SWEDISH TO BULGARIAN

Georgi Iliev

Department of Computational Linguistics
Institute for Bulgarian
Bulgarian Academy of Sciences

ABSTRACT

This paper presents a rule-based approach to the automatic translation of noun-phrases (NP) between inflectional languages, focusing on an implementation for Swedish and Bulgarian. We set out to identify possible areas of improvement of widely applied methods in the light of the present language pair, suggesting the appropriate hybrid architecture drawing on recent developments in the field. A summary of the inflectional paradigms of nouns and adjectives for the present language pair is given from a descriptive point of view illustrating the plurality of meaning of some inflectional markers and the weakness of a major application in this regard. The implementation is outlined and a detailed example of the logical approach is given, followed by an overview of the dedicated web interface. In conclusion we suggest possible areas of application of the presented method and the direction of future development.

Background

Recent work on statistical machine translation (SMT) and its commercial applications has led to significant progress but most of this work has focused on translations into English where the relatively simple morphological paradigms have compensated for the lack of linguistic sophistication of the underlying models [1]. It can be seen as a movement from a higher-dimensional (morphologically-rich) to a lower dimensional (morphologically-poor) space, where some loss of meaning and nuance is harmless [10]. On the other hand, translating from a morphologically-poor to a morphologically-rich language is especially challenging [11]. Thus pure SMT between and/or into moderately and/or highly inflected languages tends to produce 'competence' errors such as wrong use of definite markers and faulty grammatical agreement within noun phrases, among others. Although the application of phrase-based models in SMT has addressed this problem for some language pairs [10], the Bulgarian output of the most prominent translation service implementing phrase-based models, Google Translate,[1] often fails the task.

A number of studies from the late 90s on testify to the fact that substantial progress in machine translation can be achieved by combining the strengths of different approaches to machine translation in a hybrid method [2], such as SMT and example-based machine translation (EBMT) [3], rule-based machine translation (RBMT) and EBMT [1, 2], SMT and RBMT [4]. Especially the latter, in the form of integrating linguistic knowledge into SMT by way of synchronous context-free grammars (SCFG), could be considered the current state of the art in SMT [12]. However, the number of productions in a context-free grammar tends to explode if one needs to capture NP agreement in a morphologically-rich language [14].

One possible hybrid machine-translation architecture described in [4] is the feeding of SMT output into a rule-based component. [13] describes a supervised method to predict the inflected forms of a sequence of word stems derived from the output of the SMT system in order to improve accuracy over language-model generation of the target language based on language-specific morphological analysis for translation into Russian and Arabic. The probabilistic model described in [13] makes use of both monolingual and bilingual lexical, morphological and syntactic features. For the same purpose this paper presents a purely rule-based logical approach to the transfer and generation of NP agreement markers between Swedish and Bulgarian demonstrated by a suite of software programs, which could be applied in order to refine the SMT output in the absence of sufficient training data, with Bulgarian as the target language.

[1] http://translate.google.com

A common web interface is provided, which could serve as an environment for the development of transfer rules between the two languages by linguists.

Inflectional paradigms of Swedish and Bulgarian nouns and adjectives

Swedish nouns are inflected for number, definiteness and case. Swedish nouns have two grammatical genders: common (*utrum* or *en*-gender) and neuter (*neutrum* or *ett*-gender). The gender of the noun determines its definite form in both singular and plural and the form of the preceding adjective in the singular indefinite form. Adjectives agree also in number with the NP head, with a single form for both genders in the plural, such form coinciding with the definite form of the adjective [5]. Bulgarian nouns are inflected for number, definiteness and 'case' (vocative and possessive forms). Bulgarian nouns have three grammatical genders: masculine, feminine and neuter. The gender of the noun determines its definite form and the form of the preceding adjective in the singular definite and indefinite forms. Adjectives agree also in number with the NP head, with a single form for the three genders in the plural [6]. Both languages use suffixes to denote the definite form of nouns and adjectives, so that:

Swedish	Bulgarian	Swedish	Bulgarian
(**ett**) hus	къща *kušta* '*a house*'	(**en**) bok	книга *kniga* '*a book*'
Swedish	Bulgarian	Swedish	Bulgarian
hus**et**	къща**та** *kušta**ta*** '*the house*'	bok**en**	книга**та** *kniga**ta*** '*the book*'

Table 1: Suffixed definite article in Swedish and Bulgarian

A major difference between NP inflectional paradigms in Swedish and Bulgarian is that Swedish uses a separate definite article in the determiner position alongside with the suffixed definite article in cases where the head of the definite NP is modified by an adjective. In Bulgarian, on the other hand, in such cases the definite article is moved to the first modifier (unless it is an adverb), leaving the NP head in the indefinite form.

Swedish	Bulgarian	Swedish	Bulgarian
(**ett**) stor**t** hus	голям**а** къща *golyam**a** kušta* '*a big house*'	(**en**) tjock bok	дебел**а** книга *debel**a** kniga* '*a thick book*'
det stor**a** hus**et**	голяма**та** къща *golyam**ata** kušta* '*the big house*'	**den** tjock**a** bok**en**	дебела**та** книга *debel**ata** kniga* 'the thick book'

Table 2: 'Redundant' definite markers in Swedish

Areas of difficulty for SMT

Two important conclusions can be drawn from the above regarding the potential errors in the output of empirical machine translation (i.e. non-rule-based). First, that in the case of definite NP configurations in the source language, such as <DETERMINER+ADJECTIVE+NOUN>, due to the different places of the definite article in Swedish (where all three positions bear a definite marker) and Bulgarian (where two positions are realized in the surface structure and only the adjective bears a definite marker) redundant definiteness or misplacement of the definite marker is possible in the target language. Second, that agreement errors are also possible due to the ambiguity of the *-a* marker of Swedish adjectives, which could correspond to any of the following in the target language:

- singular definite form of the adjective;
- plural indefinite form of the adjective;
- plural definite form of the adjective.

The problem is augmented by the need for the adjective to agree properly with the NP head in number and/or gender.

Especially the latter errors are common in the output of Google Translate [7], as shown below:

Swedish source	Google Translate, translation into Bulgarian
De höga husen kastar stora skuggor.	Високата къщи хвърли голяма сянка.
'The tall houses cast large shadows.'	*The tall<DEF+F+S> house<PL> cast a large shadow.*

Compare:

Swedish source	Google Translate, translation into English
De höga husen kastar stora skuggor.	The high houses throw big shadows.

To a native speaker of Bulgarian the NP *'*Високата къщи*' in the output of Google Translate is strikingly ungrammatical due to the lack of agreement in number between the noun and the modifying adjective. The English NP '*The high houses*', however, does not require any agreement between the noun and the adjective, resulting in a grammatically correct translation. Furthermore, the source NP 'stora skuggor' ('*large shadows*') is in the plural, as seen from the English translation, whereas the Bulgarian translation 'голяма сянка' ('*golyama syanka*') is in the singular.

The proposed method

Five quality levels of translation are identified in [2]: indicative, informative, literal, reliable and user-oriented. A literal translation provides a translation for each unit of the source text in a correct grammatical form. The proposed method is an attempt to improve translation quality on the literal level by way of syntactic-transfer RBMT. We assume that the source text has undergone tokenization, normalization and shallow parsing to determine the boundaries of the candidate source NP.[2]

The application is organized into a source-language (morphological analysis & disambiguation) module and a target-language (generation) module. As a first step a search is performed in the source-language wordform lexicon to extract all possible morphological features of each token in the analyzed sequence, which are then converted to Prolog facts about an abstract literal uniquely identifying the token. As in most cases a token has more than one possible morphological reading, the next step is generating all possible readings of the entire sequence in the form of the Cartesian product of the readings extracted from the wordform lexicon for each token. The result is a list of sequences of Prolog facts corresponding to different readings of the NP candidate for translation, which are then asserted one by one in a Prolog database of

[2] This step has not been implemented in the application yet, therefore the need to manually enter the candidate source NP in the web interface

handwritten rules. These rules encode the intuitions the linguist has about certain constraints within NP boundaries in the source language (Swedish) in a self-explanatory manner, in particular determiner-noun and adjective-noun agreement in different NP configurations depending on the number of NP members, part of speech (POS), etc. Then a check is performed against the resulting Prolog database to determine whether any of the possible readings satisfies one of four goals describing all possible combinations of those features of the candidate NP that are relevant to translation in Bulgarian – number and definiteness. Gender, which is another feature of the entire candidate NP, is namely irrelevant to translation, because of the lack of correspondence between gender in the source language and gender in the target language.

No.	'det' → d	'stora' → e	'äpplet' → f
1	is_n(d). is_nom(d). is_pn(d). is_sg(d).	is_av(e). is_indef(e). is_nom(e). is_pl(e). is_pos(e).	is_def(f). is_n(f). is_nn(f). is_nom(f). is_sg(f).
2	is_n(d). is_nom(d). is_pn(d). is_sg(d).	is_av(e). is_def(e). is_no_masc(e). is_nom(e). is_pos(e). is_sg(e).	is_def(f). is_n(f). is_nn(f). is_nom(f). is_sg(f).
3	is_n(d). is_nom(d). is_pn(d). is_sg(d).	is_av(e). is_def(e). is_nom(e). is_pl(e). is_pos(e).	is_def(f). is_n(f). is_nn(f). is_nom(f). is_sg(f).
4	is_al(d). is_n(d). is_sg(d).	is_av(e). is_indef(e). is_nom(e). is_pl(e). is_pos(e).	is_def(f). is_n(f). is_nn(f). is_nom(f). is_sg(f).
5	is_al(d). is_n(d). is_sg(d).	is_av(e). is_def(e). is_no_masc(e). is_nom(e). is_pos(e). is_sg(e).	is_def(f). is_n(f). is_nn(f). is_nom(f). is_sg(f).
6	is_al(d). is_n(d). is_sg(d).	is_av(e). is_def(e). is_nom(e). is_pl(e). is_pos(e).	is_def(f). is_n(f). is_nn(f). is_nom(f). is_sg(f).

Table 3: Cartesian product of all possible (morphological) readings of the Swedish 'det stora äpplet' – 'the big apple'[3]

If a reading is found such that any of the four possible goals ('is_si' [singular indefinite], 'is_pi' [plural indefinite], 'is_sd' [singular definite], 'is_pd' [plural definite]) is satisfied, then the relevant goal, together with the POS-tags and the lemmas of the grammatically significant tokens for the target language (i.e. anything but the Swedish definite/indefinite article, which has no POS correspondence in Bulgarian), are passed as input to the target-language (generation) module of the application. The target-language module makes use of a list of handwritten Prolog rules to generate the required morphological features (gender, number, definite/indefinite) of each wordform in the target language (Bulgarian) for the relevant lemmas extracted from a bilingual dictionary (Swedish-Bulgarian). Similarly to the source-language analysis, the target-language generation rules encode the intuitions the linguist has about certain constraints within NP boundaries in the target language (Bulgarian) in a self-explanatory manner, in particular determiner-noun and adjective-noun agreement, and placement of the definite marker in various NP configurations depending on the number of NP members, POS, etc.

At this point we have made some strong assumptions to ensure a streamlined processing of the candidate NP, and therefore a more clear demonstration of the proposed method: first, that there will be only one valid morphological reading[4] in all possible combinations; and second, that the bilingual dictionary will provide the exact sense of the source lemma. Both assumptions are in fact irrelevant within the scope of the presented method, which aims to generate a grammatically correct interpretation of the source NP in the target language based on a set of morphological features of the source NP satisfying the underlying source-language 'grammar'. However, even in the case of homonymous words belonging to different word classes, the presented method will behave conservatively and fail if the required lemma of the relevant word class is not

[3] Prolog facts are derived directly from the tagset of the SALDO lexicon (e.g. 'is_av(e)' reads '[the token corresponding to] *e* is an "av" [adjective]')

[4] In fact the application disregards anything but the first "yes" answer to the stated goal, although multiple valid readings will be possible in Swedish, e.g. 'hans hus' could be interpreted as either 'his house' or 'his houses'

found in the morphological lexicon in the target language. Besides, in a real-life implementation of the method, which is in part intended to improve the output of SMT, we expect that the application will be provided with some partial input (e.g. stems or lemmas in the target language), which will do without the need for word-sense disambiguation.

The method is implemented in Perl and Prolog. The Swedish SALDO [8] lexicon developed by a research team at the University of Gothenburg, Sweden, is used to extract the morphological features of source-language wordforms. The Bulgarian DELAF [9] lexicon developed as part of the INTEX for Bulgarian project of the Bulgarian Association for Computational Linguistics is used to extract the required wordforms on the final generation stage.

The method is implemented as a demonstration suite of programs operating on a small set of handwritten rules and a limited bilingual dictionary because of the lack of non-proprietary bilingual resources of sufficient size to cover larger material.

An example

Source NP: det stora äpplet

'*the big apple*'

Applicable Prolog agreement and verification rules for the source NP:

```
agrees(X,Y) :- is_al(X), is_sg(X), is_n(X), is_def(Y), is_sg(Y), is_n(Y).

is_sd(X,Y,Z) :- is_al(X), is_sg(X), is_av(Y), is_a_marked(Y), is_nn(Z), agrees(X,Z).
```

One possible set of Prolog facts asserted in the Prolog database about the source NP at hand (d → '*det*', e → '*stora*', f → '*äpplet*'):

```
is_al(d). is_av(e). is_def(e). is_def(f). is_n(d). is_n(f). is_nn(f). is_sg(d). is_sg(e). is_sg(f).
```

Prolog goal satisfied by the relevant reading of the source NP:

```
?-is_sd(d,e,f).
```

Prolog facts asserted in the Prolog database about the target NP to be generated:

```
is_av(d). is_nn(e). is_sd(d,e). is_sem_f(e).
```
[5]

Applicable Prolog agreement and generation rules for the target NP:

```
is_def(X) :- is_sd(X,Y), is_av(X), is_nn(Y).

is_indef(Y) :- is_sd(X,Y), is_av(X), is_nn(Y).

agrees(X,Y) :- is_sd(X,Y), is_av(X), is_nn(Y).

is_f(X) :- agrees(X,Y), is_sem_f(Y), is_sg(Y).

is_sg(X) :- is_sd(X,Y).

is_sg(Y) :- is_sd(X,Y).
```

Target NP: голямата ябълка

golyamata yabulka

'*the big apple*'

[5] This target-language fact is asserted in addition to the facts transferred by the source-language module. It shows the gender of the noun which is a lexical-semantic feature (i.e. it is part of the entry for the relevant lemma in the wordform lexicon) and determines the form of the determiner and/or the modifying adjectives

Web interface

The method is presented as a simple, yet flexible, web interface. The user is required to enter the Swedish candidate NP and choose between different combinations of rules to be applied to analysis and/or generation. All existing Prolog rules (which have been tested and are syntactically error-free) show in read-only text areas as 'Source: existing rules' and 'Target: existing rules', respectively.

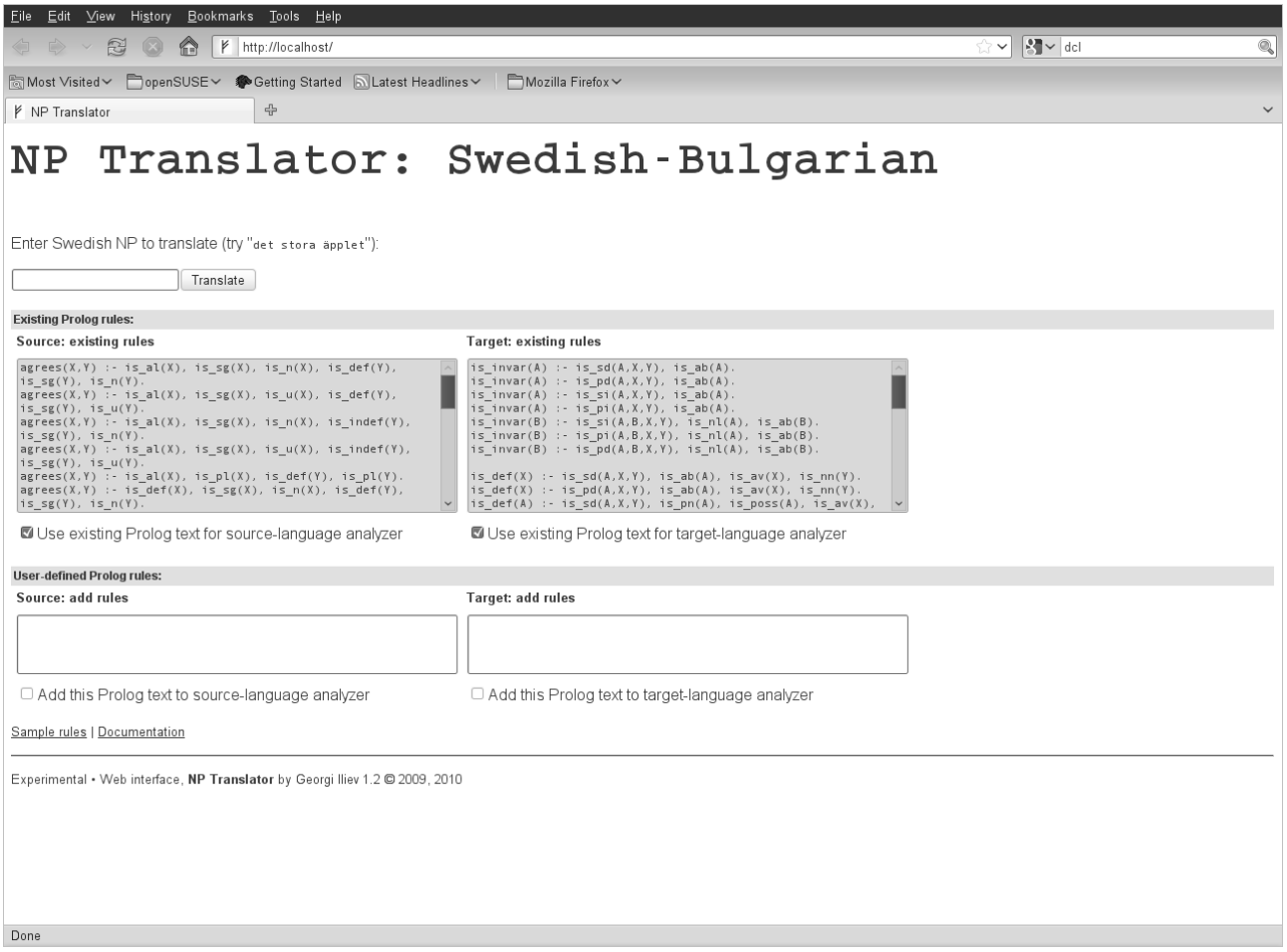


Figure 1: Web interface: successful translation of a Swedish NP using existing Prolog rules

By following the hyperlink for each candidate NP the user can view the analysis produced by the source-language module to make sure the translation is the result of the proper assignment of POS and morphology to the raw token sequence.

Figure 2: Web interface: viewing the output of the source-language module

The user can add his own rules on either side of analysis and/or generation, which can then be applied in addition to existing rules or separately (useful for testing new rules). An error message is output if a syntax error is detected in the user-defined rules, and analysis and generation continue by ignoring the erroneous user input. The application is open to extension by a virtually unlimited number of rules covering different NP configurations.

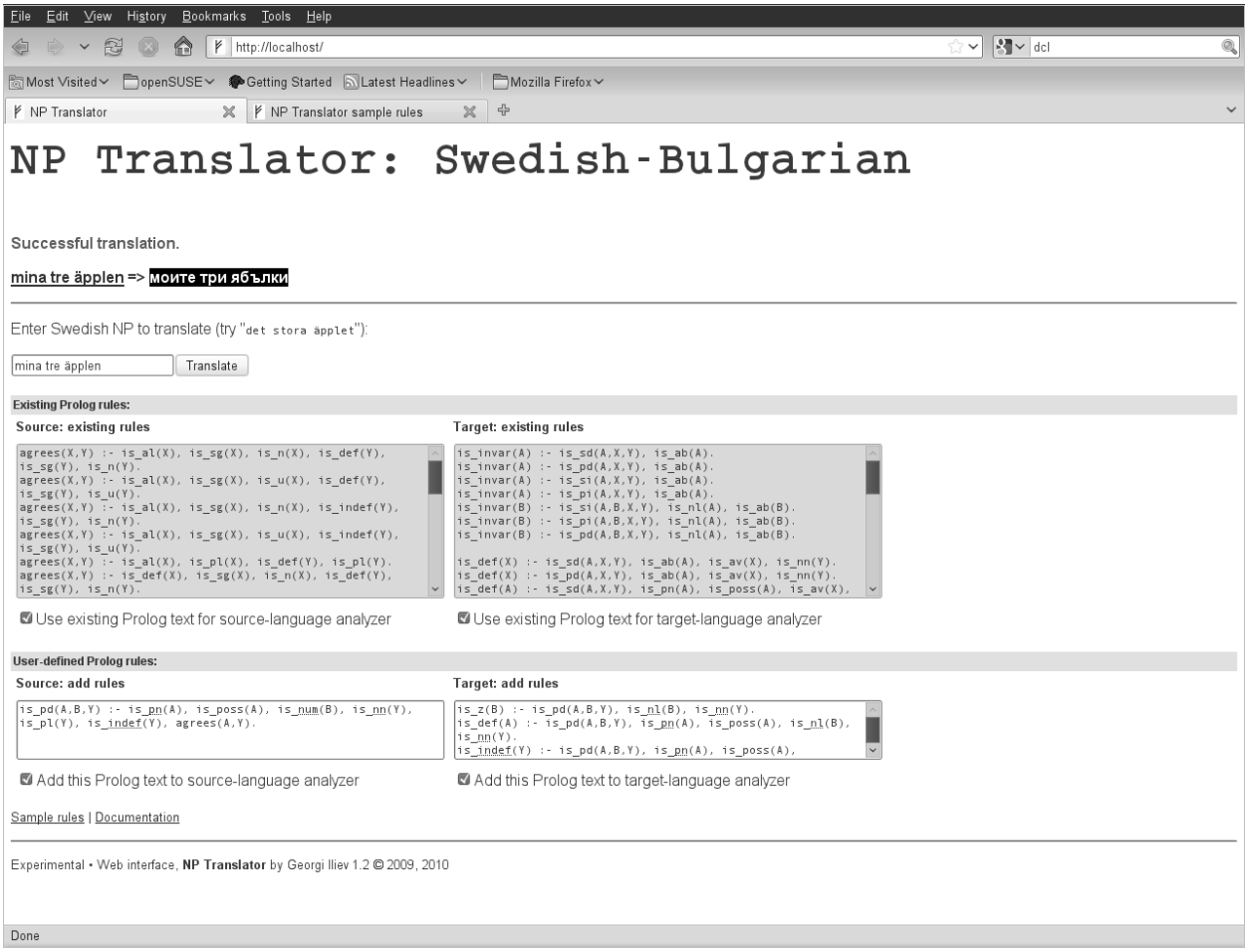


Figure 3: Web interface: successful translation of a Swedish NP using existing and user-defined Prolog rules

Possible areas of application and development prospects

On the source-language side the presented method could be applied to verify the output of constraint-grammar or probabilistic parsers, which would then allow the generation of exact agreements within NPs in the target language. On the target-language side the method could be applied to fine-tune the output of the SMT system. Promising results have been reported for tree-based SMT models employing syntactic annotation on the target side in translation to German [15]. The generation module could improve the quality of translation if combined with SMT and stemming or lemmatisation. Even when operating with a relatively small list of handwritten rules, this could do without some of the most striking errors in the output of SMT systems in the target language. A probabilistic approach to the same task is described in [13] reporting large gains in wordform prediction accuracy effective with a relatively small amount of data.

The existing rules capture only a limited number of NP configurations with preservation of word order. The next step in this respect will be the development of more complex Prolog rules to reflect some major differences in NP word order between Swedish and Bulgarian, for instance reverse word order of determiner and modifier when short forms of Bulgarian

possessive pronouns are used. We expect to be able to improve the quality of existing rules and to reduce the time spent on creating and testing new ones with the newly-developed web interface.

Swedish makes extensive use of compound nouns, which cannot be fully captured by a wordform lexicon of any size. Dealing with compounds will require a separate pre-processing step, possibly based on word frequencies [15], and a dedicated set of generation rules based on the compounds' internal syntax, which is an interesting challenge.

The lack of Swedish-Bulgarian bilingual resources currently prevents a full-fledged experiment with evaluation of the method to be performed, therefore the presented application remains a proof of concept rather than a translation tool. One possible way of completing the tool is the automatic building of a bilingual dictionary from parallel corpora. Another is the integration of the output of a probabilistic source-language parser and the output of a SMT system into the existing application to generate proper agreement within target-language NPs. Both directions of development, however, depend on the availability of the relevant bilingual resources.

Acknowledgement

This work was supported by the Mathematical Logic and Computational Linguistics: Development and Permeation (2009-2011) Project. The financial support is granted under Contract No. BG051PO001-3.3.04/27 of 28 August 2009 within the Operation Support to the development of PhD students, post-doctoral students, post-graduate students and young scientists of the General Directorate Structural Funds and International Educational Programmes with the Ministry of Education, Youth and Science.

References:

1. Eisele, A., Federmann, C., Uszkoreit, H., Saint-Amand, H., Kay, M., Jellinghaus, M., Hunsicker, S., Herrmann, T. and Chen, Y. 2008. Hybrid Machine Translation Architectures within and beyond the EuroMatrix project. In *Proceedings of the Twelfth EAMT conference*. 22-23 September 2008.

2. Carl, M., Iomdin, L., Pease, C. and Streiter, O. 2000. Towards a Dynamic Linkage of Example-Based and Rule-Based Machine Translation. Machine Translation 15. 223–257.

3. Phillips, A. and Brown, R. 2009. Cunei Machine Translation Platform: System Description. In *Proceedings of the 3rd Workshop on Example-Based Machine Translation*. Dublin, 12–13 November 2009.

4. Eisele, A. 2007. Hybrid machine translation: Combining rule-based and statistical MT systems. 1st MT Marathon, Edinburgh, April 17, 2007.

5. Holm, B. and Nylund, E. 1970. Deskriptiv svensk grammatik. Stockholm: Liber AB.

6. Koeva, S. 2001. Kratka praktičeska gramatika na bulgarskiya ezik. Sofia: Trud.

7. *Google Translate Homepage*. 2010. Google. 1 June 2010 <http://translate.google.com>.

8. Borin, L., Forsberg, M. and Lönngren, L. 2008. SALDO 1.0 (Svenskt associationslexikon version 2). Språkbanken, Göteborgs universitet.

9. Koeva, S. 1999. INTEX for Windows description of Bulgarian lexical and grammatical knowledge. In *Proceedings of "Text Corpora and Multilingual Lexicography"*. Bratislava, 4–7 November, 1999.

10. Lopez, A. 2008. Statistical machine translation. ACM Comput. Surv., 40, 3, Article 8 (August 2008).

11. Koehn, P. 2005. Europarl: A Parallel Corpus for Statistical Machine Translation. In *Proceedings of MT Summit*.

12. Hoang, H. and Koehn, P. 2010. Improved translation with source syntax labels. In *Proceedings of the Joint 5th Workshop on Statistical Machine Translation and MetricsMATR*, pages 409–417, Uppsala, Sweden, 15-16 July 2010.

13. Minkov, E., Toutanova, K., and Suzuki, H. 2007. Generating complex morphology for machine translation. In *Proceedings of the Association for Computational Linguistics (ACL)*. 128–135.

14. Jurafsky, D. and Martin, J. H. 2009. Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition, 2nd Ed. Prentice-Hall.

15. Koehn, P., Haddow, B., Williams, P. and Hoang, H. 2010. More Linguistic Annotation for Statistical Machine Translation. In *Proceedings of the Joint 5th Workshop on Statistical Machine Translation and MetricsMATR*, pages 409–417, Uppsala, Sweden, 15-16 July 2010.

ON-LINE COMPILATION OF COMPARABLE CORPORA AND THEIR EVALUATION

Radu ION, Dan TUFIŞ, Tiberiu BOROŞ, Alexandru CEAUŞU, and Dan ŞTEFĂNESCU

Research Institute for Artificial Intelligence, Romanian Academy

ABSTRACT

Using comparable corpora is became a topic in the mainstream Machine Translation (MT) research because, for less resourced languages, mining the Web for comparable corpora is assumed to be more productive than searching for parallel corpora. The experiments in using comparable corpora in enhancing translation models demonstrated significant improvements in MT accuracy. This paper reports on specific procedures of building comparable corpora from Wikipedia and from general Web using a highly customizable application that can merge diverse web crawlers and source their output either into files or NLP web services. We also describe a method of scoring a pair of documents from a comparable corpus as to their parallelism degree.

Introduction

Multilingual comparable corpora (MCC) have been around for a while in the context of Machine Translation (MT) research, as an alternative to parallel corpora which were (and still are for certain pairs of languages and domains) hard to find. By comparison with parallel corpora which contain pairs of equivalent translation units of text (sentences or paragraphs), MCC exist with different degrees of comparability: weakly comparable corpora, strongly comparable corpora, quasi-comparable corpora, very-non-comparable corpora, etc. (Skadiņa et al. 2010). A general definition of MCC that we find operational is given by (Munteanu and Marcu, 2006). They say that a (bilingual) comparable corpus is a set of paired documents that, while not parallel in the strict sense, *are related* and *convey overlapping information*. The measure of this overlapping should give the degree of the comparability between the two documents in a pair (for instance, a real number ranging between 0 and 1 with 0 indicating complete divergence of topic and 1 indicating parallelism: one document is the translation of the other).

Systematic research on methods for building and exploiting MCC is relatively new and several relevant papers can be found in the proceedings of the 1st, 2nd and 3rd workshops on Building and Using MCC: http://www.limsi.fr/~pz/lrec2008-comparable-corpora/ (LREC2008), http://comparable2009.ust.hk/ (ACL-IJCNLP 2009) and http://www.fb06.uni-mainz.de/lk/bucc2010/(LREC 2010).

There are methods of sentence alignment, named entity and terminology translation, extracting bilingual dictionaries, studying the effect of using comparable corpora on the accuracy of MT, all using comparable corpora. While the accent is naturally on the particular algorithm or model described, little or nothing special is said about the compilation of the MCC that was used. Available algorithms of collecting MCC always refer to methods to *ascertain the degree of comparability* that exists between two topically related documents. Typically, one starts with a collection of terms from a given domain along with their translation in the target language, retrieves two sets of documents corresponding to the source and the target terms and then decides which pair(s) of documents should be added to the MCC of that domain. If the sets are large (1000 documents for instance), one should have at his/her disposal a fast algorithm that will process all pairs of documents (in our example 1,000,000 documents). Therefore, having a website such as English Wikipedia in which every article is categorized and is also linked with its foreign version, be it merely a translation or otherwise a complete rethinking of the subject, constitutes an immense advantage. Thus, Wikipedia is an already established and very good quality comparable corpus[1] of many domains and the task of constructing the collection of documents pairs is greatly simplified by its structure.

In what follows, we present our MCC harvesting algorithms and applications, reporting on the sizes of corpora that have been obtained. We will sketch a scoring algorithm for computing the comparability degree of an arbitrary pair of documents, a function that is most useful in building MCC when the pairing of the documents is unknown.

Collecting Comparable Corpora from Wikipedia

For building a strongly comparable corpus one step is to identify pairs of documents that are topically related. Work in this direction is reported in Munteanu (2006) who describes a method of identifying parallel fragments in MCC. Another

[1] At least, if we speak of quality articles of this website.

experiment in pairing topically related documents is due to Tao and Zhai (2005).They tackle the problem of MCC acquisition by devising a language independent method based on the frequency correlation of words occurring in documents belonging to a given time scale. The intuition is that two words in languages A and B whose relative frequency vectors Pearson-correlate over n pairs of documents in languages A and B that are paired by a time point i, are translation equivalents. This relative frequency correlation is then used as a translation equivalence association score of words in languages A and B for describing a measure of document relatedness. Vu et al. (2009) improve the accuracy of method described above by a margin of 4% on an English-Chinese corpus. Wikipedia as a comparable corpus has been studied and used by Yu and Tsujii (2009). They sketch a simple mining algorithm for MCC, exploiting the existence of inter-lingual links between articles.

Our goal is to extract good quality MCC in languages Romanian, English and German for use in the ACCURAT project[2]. We have employed two different methods of gathering MCC from Wikipedia:

1. the first one considers an input list of good quality Romanian articles (articles that senior Wikipedia moderators and the Romanian Wikipedia community think that they are complete, well written, with good references, etc.) from the Romanian Wikipedia (http://ro.wikipedia.org/) and for each such article, it searches for the equivalent in the English Wikipedia;
2. the second one uses the Princeton WordNet and extracts all the capitalized nouns (single-word or multi-word expressions) from all the synsets. Then, it looks for Wikipedia page names formed with these nouns, extracts them and their correspondent Wikipedia pages in Romanian and German (it these exist).

The first method of MCC compilation uses 3 different heuristics of identifying the English equivalent of a given Romanian article (they are tried in the listed order):

a) it searches for an English page with the exact name as the Romanian page. For instance, we have found the following exact-match English pages (starting from the Romanian equivalents): “Alicia Keys”, “Hollaback Girl”, etc.;

b) it searches for the English link from the Romanian page that would lead to the same article in those languages. The Romanian version of the page may or may not be a complete translation from English (we noticed that the translation is usually shuffled – the narrative order of the English page is rarely kept and it usually reflects the translator’s beliefs with regard to the content of the English page);

c) it automatically transforms the Romanian page name into an English Wikipedia search query by using a translation dictionary that has scores for each translation pair. Thus, for each content word in the Romanian page name, generates the first *k* translations (*k*=2 in our experiments) and with this query, retrieves the first 10 documents from the English Wikipedia. We manually chose the right English candidate but an automatic pairing method based on document clustering is described below.

Using these heuristics, we managed to compile a very good Romanian-English comparable corpus that consists of 128 paired Romanian and English documents of approx. 502K words in English and 602K words in Romanian.

The second method of MCC compilation uses Princeton Wordnet for extracting a list of named entities. These named entities are then transformed into Wikipedia links by replacing the white spaces with underscore and adding the string “http://en.wikipedia.org/wiki/” in front of them. Then, an application performs the following steps:

a) it goes to every link and downloads the Wikipedia page if it exits;
b) every downloaded Wiki page is searched for links to correspondent Romanian and German Wiki pages; if such links exist, those pages are also downloaded;
c) all the html tags of every En-Ro or En-De pair of Wiki documents are stripped so that only the plain text remains (there is also the possibility of preserving some mark-ups for important terms highlighted in Wikipedia articles); The categories of the documents are kept in a simple database.

Using the categories of the documents one can select documents referring to specific subjects. However, due of the fact that we searched only for named entities, confusions might occur. For example, Wiki articles about Paris, Rome or London might be considered to be about sports as they are categorized, among others, as “Host cities of the Summer Olympic Games”. In reality, these articles contain very few information about such a topic. The Table 1 shows the amount of comparable data we

[2] http://www.accurat-project.eu/

extracted from Wikipedia using the described method.

Named Entities pages about:	en-ro	de-ro
Sports	1043.9 K	534.1 K
Software	63.3 K	35.8 K
Medical	617.7 K	400.9 K
Other	43,965 K	25,042.8 K
Total	**45,689.9 K (418.3 Mb)**	**26,013.6 K (239.2 Mb)**

Table 1: The amount of comparable data extracted from Wikipedia using the second method

Clustering is an unsupervised machine learning technique that groups together objects based on a similarity measure between them. This technique is appropriate for pairing documents in a comparable corpus as to their topic similarity. Classical document similarity measures rely on the supposition that the documents have common elements (words). But documents in different languages have actually very few common elements (numbers, formulae, punctuation marks, etc.) and in order to make documents in different languages similar, one approach is to replace the document terms with their equivalent translation pairs. In this approach, each document term is replaced with the translation equivalents pairs from a translation equivalents list. The document vectors for both source and target language documents are collections of translation equivalents pairs. There are several difficulties in this approach that have to be surpassed:

1. TRANSLATION EQUIVALENTS SELECTION. Not all the translation equivalents pairs have the same discriminative degree in differentiating between comparable documents.
2. CLUSTERING ALGORITHM MODIFICATIONS. The algorithm should consider pairing only different language documents.

TRANSLATION EQUIVALENTS TABLE. The accuracy of the comparable documents selection depends directly on the quality of the translation equivalents table. The translation equivalents table contains only content-word translations of lemmas with *N*-gram maximum lengths. Considering the fact that not all the translation equivalents have the same discriminative degree for selecting comparable documents, the translation equivalents table was filtered using a maximum translation equivalents entropy threshold (0.5 in our case). Using this filtering method, light verbs, nouns with many synonyms, and other spurious translation equivalents are removed.

DOCUMENT COLLECTION. The documents were tagged and lemmatized. Considering only the content words, for each n-gram from the document collection a set of translation equivalents were selected from the translation equivalents table. For example, the translation equivalents for "acetic acid" in both English and Romanian are: "acetic - acetic", "acetic acid - acid acetic", "acid - acid".

CLUSTERING FOR COMPARABLE DOCUMENTS IDENTIFICATION. This technique relies on the supposition that translation equivalents can be used as common elements that would make documents in different languages similar. We choose an agglomerative clustering algorithm. We tested several simple distance measures like Euclidean distance, squared Euclidean distance, Manhattan distance and percent disagreement. We found that percent disagreement differentiates better comparable and non-comparable documents. Considering the document vectors $x = (x_1, x_2, ..., x_n)$ and $y = (y_1, y_2, ..., y_n)$ of which elements are 1 or 0 depending on whether the corresponding vocabulary term belongs to the document or not, the percent disagreement is computed as:

$$d(x, y) = \frac{\sum_{i=1}^{n} x_i \neq y_i}{n}$$

The distance measure has the restriction that the compared documents have to be in different languages. This simple distance measure gave us a precision of 72% (with a maximum translation equivalents entropy threshold of 0.5 and a maximum of 3 translation equivalents per document term) on the collection of 128 English and Romanian Wikipedia documents described above.

Collecting Comparable Corpora from the Web

Data collection from the web is rarely a well defined job and more often than not corpus linguistics practitioners are designing their own scripts that provide an answer to the immediate need and as the problem is solved, the scripts are forgotten. Command line tools are usually applications designed to be used via text-only computer interface. We tried to give a more principled solution to reusing the small pieces of useful software and prolonging the life-time of such scripts. To this end, we developed an environment that incorporates three components: a Flow Graphical Editor which enables the user to easily

create and manage workflows, a Script Editor which assists the user in defining the processing units of the workflows and a Windows Service which takes as input the chained scripts generated by the first two components and executes the entire process at a given interval. As such, the environment is not a standalone crawler but a more general program which gives the means for high scalability and integration of modules written in different programming languages, interpreters or the use of the internal script developing system.

Out of the components described above, The Flow Graphical Editor component is the most important because it gives the advantage of graphically organizing the logic of the application around processing units and decision blocks. The user can alter the global application behavior by adding new blocks or modifying the way the output is being handled. One starts by creating the basic workflow. There are two types of active blocks: *decision blocks* and *processing units*. The Flow Graphical Editor allows for the integration of existing modules that produce console output, but the system can also enable the usage of other application types.

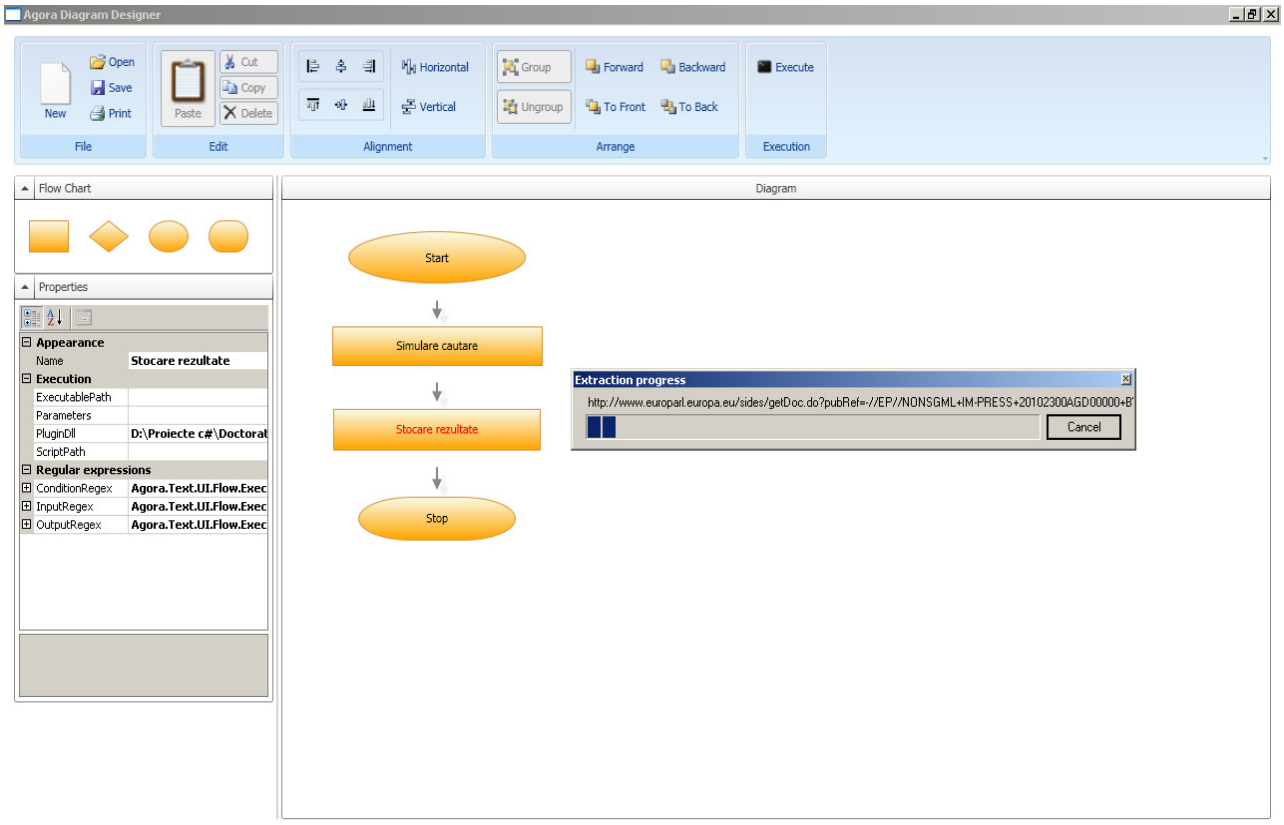


Figure 1: The Flow Graphical Editor and the execution of a diagram (a work flow)

As an example of orchestrated web crawling, we created a simple example in order to simulate the process and extract data from the European Parliament news archive. The European Parliament website provides a news section with an attached archive dating since 2004. The articles are translated 22 languages and are available for general use. The articles are classified in the sections and subsections and in order to retrieve specific articles classified accordingly, one has to perform a search using the European Parliament web interface and select its output which contains links to the desired articles.

The work flow (or diagram) for this example contains two processing units (see Figure 1): the first processing unit creates a list of articles by invoking the European Parliament web interface for searching articles in a given section and the second processing unit implements the actual data extraction which means downloading and storing the articles (provided by the first processing unit as links) on disk. At this point we can imagine a new processing unit which would feed from the output of the data extractor and would process the actual documents using the TTL web service (Tufiş et al., 2008). The design of the environment provides for this increased flexibility. If there is a need to crawl another website, one has to modify the script of the first processing unit that is responsible with collecting the links of the articles and the whole crawler is ready.

Conclusions

Mining the Web for MCC is an effective way of compensating the insufficient parallel corpora and there is a variety of different comparability levels that can be considered. Our aim is to collect MCC to enrich existing translation models. That is, we aim at extracting translation phrases (in addition to translation equivalents) from strongly MCC. To this end, we implemented several methods for strongly MCC acquisition that provided us with tens of millions of words worth of corpora. Also, we have developed a method for cross-lingually pairing documents enabling us to use the search engine gathering mechanism in order to collect strongly MCC.

Acknowledgements

The reported work has been carried within the ACCURAT project funded by the European Community's Seventh Framework Programme (FP7/2007-2013) under the Grant Agreement n° 248347.

References

Moore, R. C. (2002). **Fast and Accurate Sentence Alignment of Bilingual Corpora**. In Proceedings of the 5th Conference of the Association for Machine Translation in the Americas on Machine Translation: From Research to Real Users, pages 135–144, London, UK, 2002. Springer-Verlag

Munteanu, D. Ş., and Marcu, D. (2006). **Extracting Parallel Sub-Sentential Fragments from Non-Parallel Corpora**. In Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the ACL, pages 81–88, Sydney, July 2006. ©2006 Association for Computational Linguistics

Munteanu, D. Ş. (2006). **Exploiting Comparable Corpora**. PhD Thesis, University of Southern California, December 2006. ©2007 ProQuest Information and Learning Company

Skadiņa, I., Vasiļjevs, A., Skadiņš, R., Gaizauskas, R., Tufiş, D. Gornostay, T.: **Analysis and Evaluation of Comparable Corpora for Under Resourced Areas of Machine Translation**. In Proceedings of the 3rd Workshop on Building and Using Comparable Corpora, LREC 2010, Malta, pp. 6-14.

Tao, T., and Zhai, CX. (2005). **Mining Comparable Bilingual Text Corpora for Cross-Language Information Integration**. In Proceedings of the 2005 ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Chicago, Illinois, USA, August 21-24, 2005.

Tufiş, D., Ion, R., Ceauşu, Al., and Ştefănescu, D. (2008). **RACAI's Linguistic Web Services**. In Proceedings of the 6th Language Resources and Evaluation Conference - LREC 2008, Marrakech, Morocco, May 2008. ELRA - European Language Ressources Association. ISBN 2-9517408-4-0.

Yu, K., and Tsujii, J. (2009). **Extracting Bilingual Dictionary from Comparable Corpora with Dependency Heterogeneity**. In Proceedings of NAACL HLT 2009: Short Papers, pages 121–124, Boulder, Colorado, June 2009. ©2009 Association for Computational Linguistics

Vu, T., Aw, A. T., and Zhang, M. (2009). **Feature-based Method for Document Alignment in Comparable News Corpora**. In Proceedings of the 12th Conference of the European Chapter of the ACL, pages 843–851, Athens, Greece, 30 March – 3 April 2009. ©2009 Association for Computational Linguistics

SYNTACTIC ANNOTATION IN BULGARIAN NATIONAL CORPUS

Svetla Koeva

Department of Computational Linguistics, Institute for Bulgarian Language, Bulgarian Academy of Sciences
52 Shipchenski prohod Blvd, build. 17, Sofia 1113, Bulgaria
svetla@dcl.bas.bg

Abstract

The paper presents a data driven model for a right context-sensitive grammar construction aiming to describe as many grammatical structures as possible. The syntactic annotation in the Bulgarian National Corpus, which is fully morpho-syntactically annotated, lemmatised and linked with wordnet word senses. is provided by means of right to left cascade parsing applying a set of right context-sensitive rules. The obtained syntactic annotation can be used to facilitate the manual constituent /dependency annotation of Bulgarian Brown treebank, to exemplify different syntactic structures in Bulgarian National Corpus, and to provide data for a statistical modelling.

A general description of the Bulgarian National Corpus

The Bulgarian National Corpus (BulNC) can be described as a large-scale, general, representative and balanced corpus. The BulNC mirrors the synchronic state of the language – it covers Bulgarian texts from the middle of the XX century until the present. At this stage the Bulgarian National Corpus consists of above 320,000,000 words distributed in more than 10,000 samples (Koeva and Stoyanova 2009; Koeva et al. 2010).

The BulNC subsumes four sub-corpora: the Bulgarian Brown Corpus comprising texts from 1990 till 2005, the Structural Corpus of Bulgarian Electronic Documents from 2001 till 2010, the Structural Corpus of Bulgarian Printed Editions from 1945 till 2010, and Spoken data transcripts from 1995 till 2010. The corpus samples are drawn from varied sources: digitalised versions of previously published data, documents that are available as electronic texts[1], and some transcripts of spoken data. The BulNC is available on the Internet for collocation extraction and concordances building by means of a sophisticated search engine (Tinchev et al. 2007)[2].

Levels of annotation

The samples of the BulNC are provided with a detailed meta-data description. The meta-data description of each corpus sample (following to the great extend established standards (Atkins et al. 1992; Burnard 2007; Lee 2001) includes general and classificatory information: name of the file; information about the author (number of authors, author name(s); information about the text (number of texts, title(s), length of the sample in number of words); form of the text (written, transcribed); information about the period (date (year) of production or first publication / edition / inclusion in the corpus); information about the source (name of the publishing house, source internet address, etc.); additional notes. Following in general the Brown corpus classification, the BulNC samples are classified as informative or imaginative and further subdivided according to their domain as Administrative (Politics, Court, Education, Economy, Military, etc.), Science (Biology, Chemistry, Physics, Philosophy, Geography, etc.), MassMedia (Arts, Sport, Culture, Society, Entertainment, etc.), Fiction (Adventure, Biography, Children, Love, General, etc.) and Informal (Family, Entertainment, Children, Education, Sport, etc.) (Koeva and Stoyanova 2009; Koeva et al. 2010). The BulNC samples are also classified according to their genre (novel, story, legend, interview, dialogue, sketch, etc.).

Several levels of linguistic annotation are provided at the BulNC: morpho-syntactic by means of part-of speech tagging and lemmatisation (Koeva 2007), and semantic by means of the respective synonymous sets from the Bulgarian wordnet. The whole corpus is automatically marked up for word and sentence boundaries and automatically annotated for parts of speech, corresponding lemma and detailed grammatical information. The morpho-syntactic annotation is to the great extend unambiguous, while the semantic annotation provides an ambiguous sense analysis.

[1] At this stage the efforts are concentrated on the compilation of the electronically available documents and spoken data.
[2] http://search.dcl.bas.bg

Some parts of the corpus are manually annotated: 300,000-plus words for parts of speech and 100,000-plus words for wordnet word senses (Koeva et al. 2006). The tagset used is designed intentionally for Bulgarian language resources - it covers unambiguously and extensively the language specific morpho-syntactic properties of Bulgarian, it is flexible enough for a reduction or extension and it is convertible to other encoding conventions, i.e. MULTEXT-East[3]. The format is the so called "vertical" format - each token (i.e. word or punctuation mark) is on a separate line and the associated tags are added in tab-separated colons (different for the different level of annotation). XML output of the annotated corpus is supported as well.

During the annotation process (either automatic or manual) we observe the following principles: the input text remains unchanged (normalisation is not done); the annotation is performed consecutively (maintaining a multi-level annotation); the annotation data are represented as attribute value pairs and the annotation data are accumulated rather than overwritten. Thus even the annotation from the lowest level is retrievable, the annotation data are separable (only some parts of annotation can be accessible) and mergeable (some parts of annotation can be combined) to facilitate corpus searches (Ide and Romary 2007).

The BulNC search engine excerpts queried concordances of a given language expression and word collocations. The morpho-syntactic and semantic annotations in the BulNC (Figure 1) are exploited in order to match all the occurrences of a particular word and its forms; a word and its synonyms (or some other relation from the Bulgarian wordnet), a particular word class or grammatical category value: noun, common noun, singular, etc. and an arbitrary set of words defined by a particular set of grammatical features.

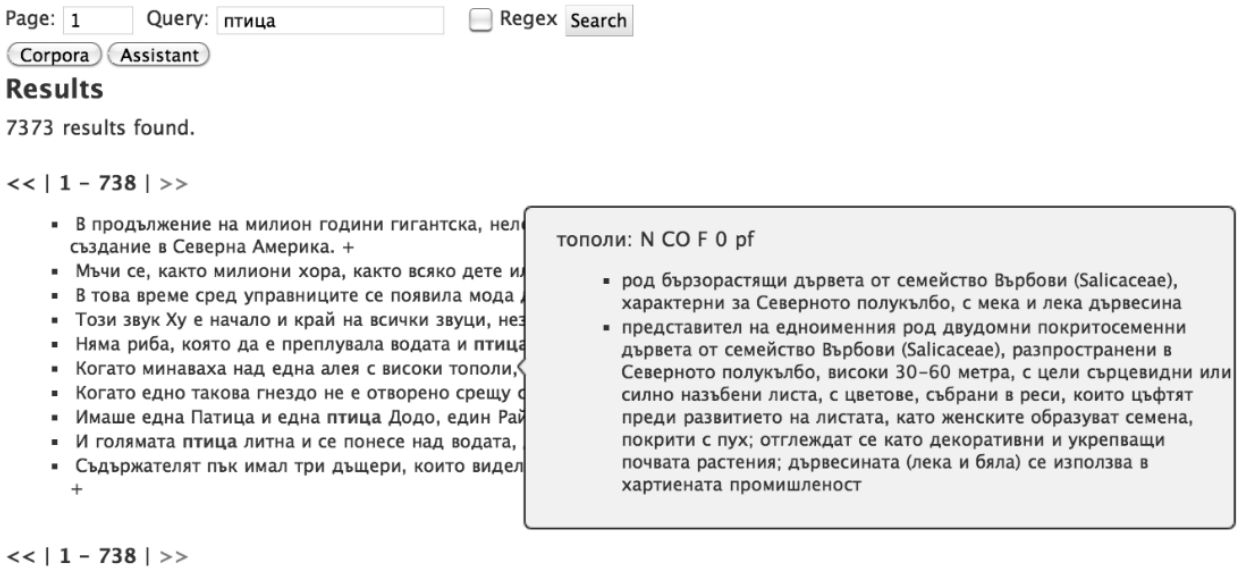


Figure 1: Morpho-syntactic and semantic annotation in the BulNC

Syntactic annotation

The ultimate goal is automatically to identify and annotate a considerable part of syntactically unambiguous constituents. We do not aim at parsing completed phrases or dependency structures rather than we intend to link contiguous words into syntactically unambiguous constituents, if a given syntactic relation exists. To achieve this goal a large amount of data exemplifying permissible sequences of word classes and grammatical values associated with a particular word form is observed and a right context-sensitive grammar is constructed. The grammar intends to identify syntactically unambiguous

[3] Despite the success of this project as a whole, shortcomings can be observed both regarding the content of the Bulgarian morpho-syntactic description (i.e. the morpho-syntactic description does not cover the minimal set necessary for the inflectional paradigm description; some of the attributes are properties of the lemma, some of them – properties of word forms only, etc.) and regarding the supporting language examples where lexical, spelling and transliteration errors occur (even in its latest version released at 2010).

constituents - namely syntactic structures that have only one syntactic interpretation, called here semi-chunk structures. Chunks are defined in terms of major heads as follows (Abney 1991):

Let h be a major head. The root of the chunk headed by h is the highest node in the parse tree for which h is the s-head, that is, the 'semantic' head. S-heads can be defined in terms of syntactic heads, however, as follows. If the syntactic head h of a phrase P is a content word, h is also the s-head of P . If h is a function word, the s-head of P is the s-head of the phrase selected by h.

By this definition, chunks are non-recursive (never containing a phrase of the same category as itself) which expand from the left periphery of a phrase to the phrasal head (Abney 1991). Abney-style chunk parsing is implemented as cascaded, finite-state transduction (Abney 1996) which allows a chunk to contain other chunks.

The semi-chunk constituents in focus differ from the traditional chunks by the following main properties: no difference between syntactic and semantic heads is considered, expansion is from the right periphery to the phrasal head embracing largest syntactically unambiguous constituent, right-hand modifiers are allowed. A simple context-free grammar is proved to be adequate to describe the structure of chunks (Abney 1991), while a right context sensitive grammar is applied for semi-chunks.

Some sets of context-free phrase-structure rules are proposed in Bulgarian grammars (Penchev 1993; Koeva 1999 among others) which describe the general dependencies in the language such as constituency, heads etc., but do not give detailed information for real combinatory properties of word classes. Some characteristics of Bulgarian nominal chunks (Osenova 2002) as well as relatively detailed grammar of Bulgarian simple sentence (Petrova 1999) have been also proposed, but published descriptions are neither enough complete nor enough consistent.

In order to provide a comprehensive and consistent description we combine a corpora-driven analysis with extensive grammatical information to construct a model for grammar rules compilation. The trigrams calculated over the BulNC are extracted and unified (the trigram language model corresponds to binary context-sensitive rules with a defined right context). Each term constituting the trigram is a "vertical" format separate line, containing the triple 1) word form and associated tags - 2) lemma and 3) annotation for the grammatical class of the lemma plus the values of the grammatical categories of the word form (punctuation is also considered). The trigrams are divided in different groups according to the second term word class and sorted after the third term word class.

For example the sentence *Израелската армия отстрани от своите редици трима запасни офицери, които подписаха петиция, че отказват да служат в окупираните палестински територии (The Israel army removes from its ranks three reverse officers who was signed a petition (claiming) that they refuse to serve in the occupied Palestinian territories)* is split in 21 trigrams (Table 1).

word form	**lemma**	**annotation**	translation	**word form**	**lemma**	**annotation**	translation
Израелската	Израелската	A---:sfd	The Israel	петиция	петиция	NCF-:sf0	a petition
армия	армия	NCF-:sf0	army	,	,	PU	
отстрани	отстрани	V2PT:R3p	removes	че	че	JS--	that
от	от	PREP	from	отказват	отказвам	V2IT:R3p	(they) refuse
редиците	редица	NCF-:pfd	the ranks	да	да	JE--	to
си	свой	PH--:z	its	служат	служа	V2II:R3p	serve
трима	три	CK--:w0	three	в	в	PREP	in

запасни	запасен	A---:pm0	reserve	окупираните	окупирам	V2II:Qpd	the occupied
офицери	офицер	NCM-:pm0	officers	палестински	палестински	A---:p0	Palestinian
,	,	PU		територии	територия	NCF-:p0	territories
които	който	PGB-:p	who	.	.	PU	
подписаха	подпиша	V2PT:E3p	have signed				

Table 1. Example of extracted trigrams

The goal is to define what are contiguous combinatory properties of the extended word classes. When the trigrams extracted from the example sentence are grouped it is seen that fife trigrams contain a noun as a second term (Table 2). The reordering against the third term helps to identify which right contexts unambiguously mark the phrase borders, for example the last term in the trigram *служат в окупираните (served in the occupied)* is not a phrase delimiter; to decide whether first and second terms built a phrase, for example in the trigram *израелската армия отстрани* (*the israel army removes*) the third term is a phrase delimiter and the first and the second terms built a noun phrase); to decide which term (first or second) is the phrase head; and to exploit and generalise this information in grammar rules.

word form	**lemma**	**annotation**	**word form**	**lemma**	**annotation**
Израелската	Израелската	A---:sfd	запасни	запасен	A---:p0
армия	**армия**	**NCF-:sf0**	**офицери**	**офицер**	**NCM-:p0**
отстрани	отстрани	V2PT:R3p	,	,	PU
подписаха	подпиша	V2PT:E3p	палестински	палестински	A---:p0
петиция	**петиция**	**NCF-:sf0**	**територии**	**територия**	**NCF-:p0**
,	,	PU	.	.	PU

Table 2. Grouped and sorted trigrams

Further, among the group of trigrams with a noun second term, the third term is distinguished as a phrase border in four cases (being a punctuation or a verb) (Table 2). The first level right context-sensitive rules describing these cases, are as follows:

1NCF-:sfdP → A---:sfd **NCF-:sf0** RC=V

1NC[MF]-:pm0P → A---:pm0 **NC[MF]-:pm0** RC=PU

The properties of syntactic relations formalised in the context-sensitive rules are described according to the following principles. The syntactic relations hold either between individual words, or between a word and a phrase, both building a sentence constituent. Only binary relations linking two and only two constituents are taking into consideration. If there is a phrase, the phrase might be in any position at the left-hand part of the rule. The rules are context dependent such as only negative right context is considered. The format of the rules do not allowed overlapping. The syntactic relations in focus can be described as inverse, asymmetric and areflexive - each pair links one head and one dependent (applicable to the coordination as well), some of the relations are transitive. The parsing operates from right to left in a cascade manner giving

a priority to word to word linking and continuing with word to phrase combinations, exploiting all rules in the grammar such as the output of a rule might become the input of another rule.

The constituted rules are applied to the corpus and second level trigrams (where a trigram term can be a phrase) are extracted such as the already picked phrase delimiter terms being always the third term in the trigrams (Table 3).

word form / annotated phrase	**lemma**	**annotation**	**translation**	**word form / annotated phrase**	**lemma**	**annotation**	**translation**
трима	три	CK--:w0	three	окупираните	окупирам	V2II:Qpd	the occupied
запасни офицери		**1NC[MF]-:pm0P**	**reserve officers**	**палестински територии**		**1NC[MF]-:pm0P**	**Palestinian territories**
.	.	PU		.	.	PU	

Table 3. Second level trigrams

The same procedures of trigrams grouping and sorting are applied. If applicable, new rules might be constituted to describe the constituency, in our example the new rules are:

2NC[MF]-:pm0P → CK--:w0 1**NC[MF]-:pm0**P RC=PU

2NC[MF]-:pmdP → V2II:Qpd 1**NC[MF]-:pm0**P RC=PU

The steps of rules application over the corpus, trigram extraction, analysing the trigrams content and writing new rules might be repeated until all cases where the first and the second terms constitute a phrase are processed.

For example the grammar for the example sentence is:

1NCF-:sfdP → A---:sfd **NCF-:sf0** RC=V

1NC[MF]-:pm0P → A---:pm0 **NC[MF]-:pm0** RC=PU

1NC[MF]-:pm0P → NCF-:pfd PH--:z RC= CK--

1 V2PT:E3pP → V2PT:E3p **NCF-:sf0** RC=PU

2 PREPP → PREP **1NC[MF]-:pm0** RC= CK--

2NC[MF]-:pm0P → CK--:w0 1**NC[MF]-:pm0**P RC=PU

2NC[MF]-:pmdP → V2II:Qpd 1**NC[MF]-:pm0**P RC=PU

2V2PT:E3pP → PGB-:p V2PT:E3pP RC=PU

3PREPP → PREP **2NC[MF]-:pmd** RC=PU

3V2II:R3p P- → **V2II:R3p** 2PREPP RC=PU

The rules can be identified with a very high accuracy (if we consider word class to word class relations). Some of the general classes of rules are for example: an adjective immediately followed by an indefinite noun or by an indefinite noun phrase constitute an adjectival noun phrase, a pronoun (possessive, reflexive, relative, negative, collective, indefinite) immediately followed by an indefinite noun or by an indefinite noun phrase constitute a pronominal noun phrase, an ordinal numeral immediately followed by an indefinite noun or by an indefinite noun phrase constitute a numeral noun phrase, and so on. Applying the gender, number and definiteness agreement to each of the above listed rules results in 72 different rule variants. To summarise, we intent to cover as many as possible grammatical sequences that constitute a phrase - thus, following the Zipf's law: few rules occur very often, while many rules occur relatively rare.

Within the Bulgarian National Corpus, the syntactic annotation is applied in a cascade manner, utilising the level of morpho-syntactic annotation and applying an extensive right context- sensitive grammar. The evaluation of the automatic parsing is

performed manually over a relatively small part of the corpus - app. 150 000 words. The results of the evaluation answered to the expectations: some of the annotated phrases coincide with multi-word expressions, word order dependencies are not parsed, a limited number of named entities is recognised as well.

Conclusion and future work

The annotation in BulNC is not limited to a particular level and / or particular categories - our aim is to provide a reliable linguistic annotation answering to different research tasks. The following levels of linguistic annotation are further considered for application: analytical verb forms, multi-word terms, named entities and some idiomatic expressions. The unambiguous syntactic annotation can be used for several different purposes: to facilitate the manual constituent /dependency annotation of Bulgarian Brown tree bank - part of the Bulgarian Brown corpus, to recall and exemplify different syntactic structures in BulNC, and to provide data for statistical modelling of the syntactic structure of Bulgarian.

References

Abney 1991: Abney, S. Parsing by chunks. In: R. Berwick, S. Abney, and C. Tenny, editors, *Principle-Based Parsing*, pp. 257–278. Kluwer.

Abney 1996: Abney, S. Partial parsing via finite-state cascades. In: John Carroll, editor, *ESSLLI Workshop on Robust Parsing*, pp. 8–15, Prague, Czech Republic.

Atkins et al. 1992: Atkins, B.T.S., Clear, J., and Ostler, N. , Corpus Design Criteria, In: *Literary and Linguistic Computing,* 7, pp. 1-16.

Burnard 2007: Burnard L. (ed) *Reference Guide for the British National Corpus* (XML Edition) http://www.natcorp.ox.ac.uk/docs/URG/

Ide and Romary 2007: Ide, N. and L. Romary. Towards International Standards for Language Resources. In Dybkjaer, L., Hemsen, H., Minker, W. (eds.), Evaluation of Text and Speech Systems, Springer, pp. 263-84.

Koeva 1999: Koeva, Sv. *Syntax and punctuation*, Plovdiv.

Koeva 2007: Koeva, Sv. Multi-word Term Extraction for Bulgarian, ACL 2007, In: *Proceedings of the Workshop on Balto-Slavic NLP*, pp. 59-66.

Koeva and Stoyanova 2009: Bulgarian National corpus, In: *Bulgarian language*, № 3, pp. 137-145.

Koeva et al. 2006: Sv. Koeva, Sv. Leseva, I. Stoyanova, E. Tarpomanova, M. Todorova. Bulgarian Tagged Corpora. In: *Proceedings of the Fifth International Conference Formal Approaches to South Slavic and Balkan Languages*. Sofia, pp. 78-86.

Koeva et al. 2010: Koeva Sv., D. Blagoeva and S. Kolkovska. Bulgarian National Corpus Project, In: N. Calzolari, K. Choukri, B. Maegaard, J. Mariani, J. Odjik, S. Piperidis, M. Rosner, D.Tapias (eds.) *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10)*, 2010, Publisher: European Language Resources Association (ELRA). ISBN 2-9517408-6-7

Lee 2001: Lee, D. 'Genres, registers, text types and styles: clarifying the concepts and navigating a path through the BNC Jungle', In: *Language Learning and Technology*, vol 5 no 3, September 2001; http://llt.msu.edu/vol5num3/lee/default.html

Osenova 2002: Osenova, P. Bulgarian Nominal Chunks and Mapping Strategies for Deeper Syntactic Analyses. In: *Proceedings of The First Workshop on Treebanks and Linguistic Theories* (TLT2002), 20th and 21st September 2002, Sozopol, Bulgaria. pp. 150-166.

Petrova 2009: Petrova Iv. *Syntactic analysis of Bulgarian simple sentence*, PhD Thesis, Sofia.

Penchev 1993: Penchev J. *Bulgarian syntax. Government and Binding*, Plovdiv.

Tinchev et al. 2007: Tinchev, T., Sv. Koeva, B. Rizov, N. Obreshkov. System for advanced search in corpora. In: *Literature and writing in Internet*, St. Kliment Ohridski University Press, Sofia, pp. 92-111.

BULGARIAN SENSE-ANNOTATED CORPUS – RESULTS AND ACHIEVEMENTS

Svetla Koeva, Svetlozara Leseva, Ekaterina Tarpomanova, Borislav Rizov, Tsvetana Dimitrova, Hristina Kukova

Department of Computational Linguistics, Institute of Bulgarian Language, Bulgarian Academy of Sciences
52 Shipchenski prohod blvd., bldg. 17, 1113 Sofia, Bulgaria

ABSTRACT

The paper offers a discussion of the principles and practicalities behind the Bulgarian Sense-Annotated Corpus (BulSemCor), presents the results, and sketches the challenges encountered in the process of annotation, the adopted conventions and the decisions made. First, the corpus structure and the tool for annotation are presented in brief, followed by a discussion of the methodology for identification and annotation of different types of language units, the strategies towards challenging phenomena with respect to part-of-speech and morpho-syntactic classification, the approaches for handling certain syntactic phenomena such as elliptic constructions and coordinate compound words, etc. The encoding of language-specific concepts and the decisions with respect to the organisation of BulNet (the lexical-semantic net that provides the inventory of senses for annotation), are also covered. Finally, the corpus applications and future developments are outlined.

Introduction

The Bulgarian sense-annotated corpus (BulSemCor) has been developed according to the methodology underlying the SemCor corpus created at Princeton University (Fellbaum et al. 1998; Landes et al. 1998, Miller et al. 1993). It is a subset of the Brown Corpus of Bulgarian (BCB) sense-tagged according to the Bulgarian wordnet (BulNet) (Koeva et al.2006). Unlike SemCor in which only open class words are POS-tagged, lemmatised and sense-annotated, BulSemCor adopts an 'exhaustive' annotation strategy, that is, not only content words, but also function words, numerical expressions, etc. are subject to annotation. In consequence, apart from annotation proper other tasks needed to be specified and carried out.

Corpus structure and representation

BulSemCor is composed of 811 text excerpts of 100+ running words each, adding up to 101,768 tokens. The corpus preserves the original structure of the reference corpus by including an excerpt from each BCB unit sampled according to the density of high frequency open-class lemmas.

BulSemCor is represented in an XML format that makes use of a flat data structure. Tokens are encoded in tags named word whose attributes store relevant information such as form, lemma, sense annotation, annotator's name, time stamp. Another attribute encodes a parent ID that links the tokens identified as part of a compound.

The pre-processing of the corpus involved lemmatisation using the Grammar Dictionary of Bulgarian (Koeva 1998) and POS-tagging.

In the course of annotation language units (LU) – one or more tokens denoting a single concept – are identified. The lemma of each LU is mapped to the BulNet literals (members of synsets) having the same lemma, thereby associating the LU with the corresponding senses available in BulNet. BulNet's overall structure corresponds to that of Princeton WordNet 2.0. (PWN 2.0), but language-specific concepts and features have been introduced, as well. It currently consists of over 32,000 synonym sets representing lexicalised concepts. Each synset is supplied with a gloss, and optionally with usage examples, notes regarding the grammatical form, usage, etc. of the literals and/or the synset. BulNet is stored as an XDB convertible MySQL database that may be accessed either locally, or on the web.

Corpus annotation tool

The annotation tool Chooser (Koeva et al. 2008) makes use of centralised data storage and affords concurrent user access. The UI has a tripartite display area: a main pane – where the corpus is loaded, a list view pane in which the available annotation options are viewed, browsed and selected, and info view that embeds the visualisation modes of the wordnet development tool. Chooser allows different navigation strategies including passes of (i) all tokens; (ii) all instances of a token; (iii) annotated tokens; (iv) markables, as well as operations over tokens – (i) delete, insert and edit functions; (ii) selection of MWEs, including non-contiguous constituents.

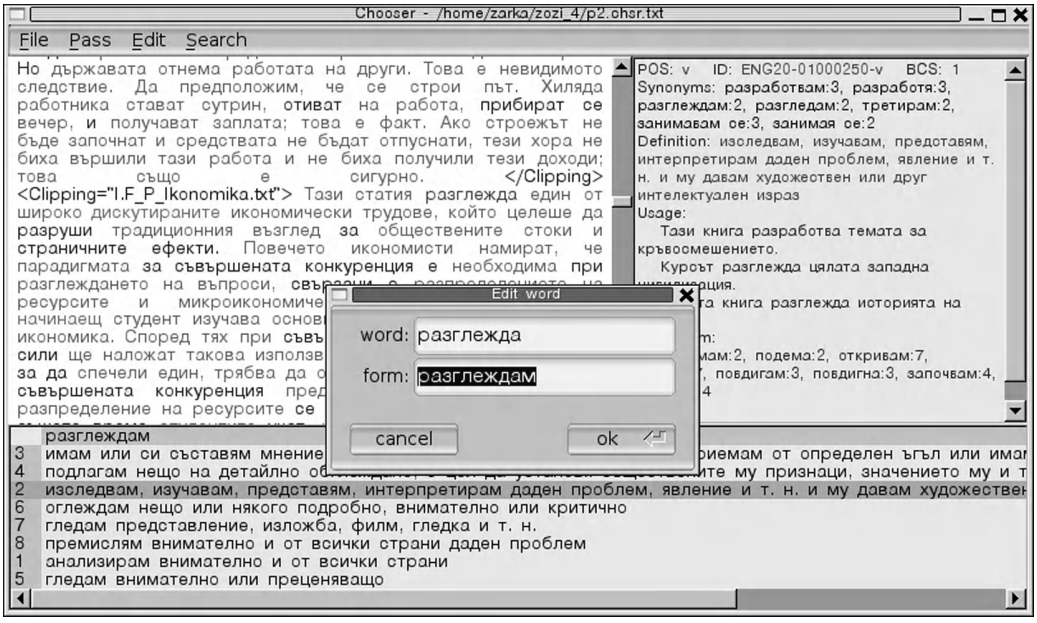


Figure 1: Chooser

The info view and the list view supply the relevant information (selection options, visualisation of synsets) from BulNet. The communication model provides a uniform interface with dynamic updates of changes in BulNet that are made available to annotators at run-time. The tool is implemented in Python and is OS-independent (tested on Linux, Windows and MAC OS).

The corpus – current state and achievements

BulSemCor's annotation may be considered in terms of corpus annotation proper and wordnet expansion and enhancement.

A chief prerequisite for the appropriate sense annotation or other kinds of token markup proved to be the accurate delimitation and identification of language units – (i) single words; (ii) multi-word expressions including compounds, idiomatic expressions, etc.; (iii) named entities including names, numerical and time expressions.

Another important consideration has been the proper morpho-syntactic classification of identified LUs. Generally, single and compound words found in the corpus are classified into 9 parts of speech, corresponding to the traditional ones, except for numerals, which are encoded in the PWN (and hence in BulNet) as adjectives or as nouns, depending on their morphological properties and syntactic function – as modifiers or as heads of phrases. In the course of annotation the morpho-syntactic approach has been extended to other cases. Most notably a similar convention has been adopted for words classified as adverbs in the Bulgarian grammatical tradition that may modify both NPs (therefore encoded in Bulnet as adjectives and annotated respectively in BulSemCor) and VPs (encoded and annotated as adverbs), including adverbial-like MWEs which are not at all classified in terms of POS under traditional accounts. Thus, *na zhivo* is annotated as an adjective (corresponding to PWN synset *{live:9; unrecorded:1}* in *predavane na zhivo* (live broadcast), and as an adverb (corresponding to PWN synset *{live:12})* in *Predavaneto shte bade izlachvano na zhivo (The programme will be broadcast* live).

Other considerations determining the choices made by annotators refer to the identification of: (i)substantives; (ii) elliptic constructions including ellipsis in coordinated free phrases; (iii) ellipsis in coordinating compound words; (iv) constructions.

Following the PWN methodology substantives are encoded in BulNet as nouns and are annotated with the relevant sense, e.g. *bolniya* in *V stayata na bolniya sveteshe* (The room of the sufferer was lit) is annotated as *{bolen chovek:1; bolen:2 [subst. adjective]; stradasht [subst. participle]}* corresponding to *{sick person:1; diseased person:1; sufferer:2}*.

The non-omitted constituent of elliptic NP phrases is annotated according to its own part of speech and meaning. This

approach is adopted in the annotation of coordinated phrases, as well, e.g. in *novata i starata kniga* (the new and the old book) *novata* (new) and *starata* (old) are identified and annotated separately as adjectives ("l=" stands for the lemma, "w=" stands for the word as it appears in the corpus):

<word l="***nov***" s="10015896421" w="***novata***"/><word l="i" s="104137004110" w="i"/><word l="***starata***" s="10015872451" w="***starata***"/><word l="***kniga***" s="10060130913" w="***kniga"***/>

An element of a compound word may also be omitted. This is typically the case when two or more compound words are coordinated. Unlike elliptic phrases, compounds denote distinct concepts and should be appropriately identified and annotated. Further support in favour of this approach comes from interlingual rendition, consider the example: *pryasno i kiselo mlyako.* In English 'milk' (*pryasno mlyako*), and 'yogurt' (*kiselo mlyako*) are not compounds. Therefore, the non-omitted constituent of the compound is re-lemmatised by the annotator according to its main entry form:

<word l="***pryasno mlyako***" s="10073702283" w="***pryasno***"/><word l="i" s="104137004110" w="i"/><word l="kisel" p="-1525323956" pl="***kiselo mlyako***" s="10073752293" w="***kiselo***"/><word l="***mlyako***" p="-1525323956" s="10073752293" w="***mlyako***"/>

Certain constructions such as *in ... context*, *in ... aspect* where the slot may be filled by a number of different words – this/similar/such, etc. are annotated as separate entities. Consistency of the senses assigned to the elements of the constructions throughout the corpus is secured by an inter-annotator agreement.

Results

1. At present, the annotation of 90% of the corpus has been completed. MWEs add up to 6.57% of the total number of annotated LUs. The difference between the number of the annotated tokens – app. 89,000, and that of the annotated LUs – 81,876, is due to the fact that the elements of a MWE are counted as one LU.

2. The adopted strategy to mark up all the words has resulted in a lemmatised POS and sense-annotated corpus of units of running text. Semantic annotations cover both open-class and function words. While in compliance with the overall wordnet structure, sense distinctions in the closed word classes have been drawn primarily from corpus evidence.

Annotated LUs according to POS									
POS	Nouns	Verbs	Adj	Adv	Preps	Conj	Pron	Part	Interj
Number	25686	13166	10525	5629	12448	6040	5982	2391	9

Table 1: POS distribution of annotated LUs

3. In the course of identification and annotation of LUs in BulSemCor, a strategy for encoding language-specific concepts such as productive derivation patterns not characteristic of English (and in consequence not available in the PWN) has emerged and an encoding specification compliant with the wordnet structure has been adopted. The latter includes (i) encoding a LU in an existing synset where this is appropriate, or (ii) creating a new synset where no corresponding sense is lexicalised.

3.1. Feminine nouns. Bulgarian female agent or occupation nouns, etc. corresponding to gender-neutral nouns in English are added as synonyms to synsets containing the respective male representative after taking into consideration the relevant sense, for example *sekretarka* is added to the synset *{sekretar:1} – {secretary:3, secretarial assistant:1}* – 'an assistant who handles correspondence and clerical work for a boss or an organization', and not to the synset *{sekretar:2}* - *{secretary:4}* – 'a person who is head of an administrative department or government'. A synset note is created explaining that the two lexical items are subsumed by the sense of the synset.

3.2. Substantives. Substantivised words denoting distinct concepts, mostly adjectives and participles, are encoded either as synonyms in existing synsets or as hyponyms of appropriate superordinate nouns synsets.

3.3. Diminutives and augmentatives. The derivation of diminutives (and to a lesser extent of augmentatives) by adding a suffix to neutral nouns is a productive pattern of Bulgarian. Therefore, the following criteria for encoding diminutives and augmentatives found in the corpus have been adopted:

(i) there is an equivalent in PWN;

(ii) the diminutive/augmentative denotes a distinct concept from the noun it derives from, e.g. *{brada:3}* – *{beard:3}* – 'the hair growing on the lower part of a man's face' and *{brada:1, bradichka:1}* – *{chin:1}*;

(iii) it is a part of a compound noun or a set phrase, in which it cannot be transformed to the noun it is derived from;

(iv) the diminutive or the augmentative is consistently found in language interchangeably with the neutral noun without change of meaning or connotation.

If the criteria (i-iii) are met, the diminutive/augmentative is added to the BulNet equivalent of the corresponding PWN synset. If criterion (iv) is met – the lexical item is encoded as a literal of the synset containing the noun it is derived from: *{momiche:1; momichentse:1}* – *{female child:1; girl:2; little girl:1}* – 'a youthful female person'. When there is no PWN synset that lexicalises the concept, a new one is created in BulNet as a hyponym of an appropriate superordinate synset. When none of the criteria is applicable, the word is re-lemmatised as the form of the noun it is derived from.

3.4. Relational adjectives represent a productive derivation pattern in Bulgarian that embraces both common and proper nouns and a variety of relations with respect to the noun such as material, action, time, place, person, number, etc. In the general case they do not have correspondences in English, which instead employs nouns modifiers, and are hence encoded as new synsets and linked to the noun they are derived from by means of the [derived] relation, e.g.: *{studentski:1}* (student's) to *{student:1}* (*{college student:1; university student:1}*).

3.5. Compound adjectives and adverbs. Adjectives and adverbs may be formed by combining two stems. Following the adopted principle in the PWN, Bulgarian specific synsets are encoded in BulNet as new synsets and connected within the wordnet structure by means of the relation [derived] to the adjective or noun they are derived from for compound adjectives, e.g. *{socioeconomic:1}* to *{economic:4}*, however *{sociolcultural:1}* to *{society:1}* and *{acculturation:3, culture:4}*, or to the corresponding adjective for adverbs – *{socioeconomically:1}* to *{socioeconomic:1}*.

3.6. Verbs. Verb classes that do not have English correspondences are encoded in BulNet as hyponyms in an appropriate semantic class tree:

(i) phase (inceptive, completive) verbs, e.g. *{zapyavam:1; zapeya:1}* ('begin singing'), *{dostroyavam:1; dostroya:1}* – 'finish building', are encoded as hyponyms of both the verb they are derived from (e.g. sing, build) and the verb synset denoting the phase;

(ii) (temporal, degree, path, etc.) prefixed measure verbs (Filip 2008) including spatial, directional, etc. measure functions. Prefixed verbs that do not shift to a different semantic class, such as temporal or degree *po*-verbs *{pomalchavam:1; pomalcha:1}* – 'be silent for a while'; *{poozdravyavam:1; poozdraveya:1; povazstanovyavam se:1; povazstanovya se:1; posavzemam se:1; posavzema se:1}* – 'recover to a certain extent from illness or shock' - are encoded as hyponyms of the verbs they are derived from. Verbs that shift to a different semantic class as compared with the simplex are encoded as hyponyms in the respective class, e.g. *{otlitam:1; otletyavam:1; otletya:1;}* – 'fly away' is encoded under *{tragvam:1; tragna:1; zaminavam:1; zamina:1;}* – *{leave:8; go forth:2; go away:3}*.

(iii) state and inchoative verbs formed from adjectives corresponding to the English constructions *be*+*Adj*, *become/get/grow/go* + *Adj*, respectively, e.g. *{beleya se:1}* – 'be/stand out white', *{oglushavam:1:oglusheya:1}* – 'become/go/grow deaf'. Like other inchoative change of state verbs they are encoded as hyponyms of the synset *{promenyam se:1; promenya se:1}* – *{change:11}* – 'undergo a change'.

(iv) lexicalised reflexive verbs corresponding to English reflexive readings of transitive verbs – *{obrazovam se:1}* – 'educate oneself', *{emantsipiram se:1; osvobozhdavam se:4; osvobodya se:4}* – 'liberate oneself'. Such verbs are linked to appropriate hypernyms after taking into consideration the semantic class they belong to. The above examples refer to change of state verbs, and are therefore encoded in the hypernym tree of *{change:11}*.

3.7. Closed-class words. Closed-class words are integrated into the wordnet structure through the [category_domain] relation which links them to the synset denoting their categorial classification as *{preposition:1}*, *{particle:1}*, *{coordinating conjunction:1}*, *{subordinating conjunction:1}*, and *{function word:1, closed-class word:1}*.

Prepositions' senses have been defined in such a way as to account for the variety of their formal (linking the arguments to the predicate) and predicative (expressing adverbial-like senses) functions (Nitsolova 2008).

Conjunctions fall into two classes depending on the type of sentence they introduce – coordinating and subordinating, accounting for the distinct relations between phrases and clauses.

Particles are classified in terms of their function – negation, reflexivity, modality, interrogativity, among others. The particles expressing intensification are integrated through the [usage_domain] relation linking them to the synset *{intensifier:1}*. Additionally, negation particle *{not:1}* is connected to its antonym *{yes:1}* through the [near_antonym] relation.

POS	**P**	**Conj**		**Particle**	**Ij**
		Coordinating	Subordinating		
Synsets	374	49	52	54	5
Literals	593	101	106	70	5

Table 2: Function words

3.8. Culture-specific and language-specific words including person names, place names, etc. (Koeva et al. 2006)

4. BulSemCor incorporates a small knowledge base of annotated NEs and NE patterns including embedded NEs. Unlike the names of famous persons, places, organisations, brands, etc. which are encoded in PWN and/or in BulNet, generally proper names are lemmatised and annotated with the synset corresponding to the most relevant NE ontological category. In the following example the first name *Nikolay* has been lemmatised as sobstveno ime (given name), *Ivanov* – as familia (family name), and the two are annotated accordingly:

<word l="***sobstveno ime***" s="10059465433" t="None" u="None" w="***Nikolay***"/><word l="***familia***" s="10059461403" t="None" u="None" w="***Ivanov***"/>

Wordnet development

In parallel to the development of BulSemCor BulNet has been expanded to up to 32,000 synsets, an increase by approximately 50%, and enhanced with new features. Apart from the classes identified in the previous section, several other specifics of BulNet are worth mentioning.

In creating language-specific synsets sense distinctions and definitions have been largely knowledge-base driven and corpus-based in that BulSemCor instances have been analysed with consideration to English and Bulgarian lexicographic resources. At the same time the general principles underlying the wordnet structure have been respected. Additionally, the majority of the newly-created synsets have been supplied with translation equivalents.

Certain optimisations have also been made – adverbs have been classified into 18 ontological categories, the most numerous of which are manner (355 synsets), time (155 synsets), and location (68 synsets), and linked to the synset of the relevant ontological category through the [category domain] relation and/or to a synset denoting their function such as adjunct (48 synsets) and intensifier (24) through the [usage domain] relation. Pronouns have been classified in 9 classes (personal, possessive, reciprocal, demonstrative, interrogative, relative, indefinite, negative, generalising) and linked to the respective synset through [category domain].

Applications

The corpus is indispensable for any linguistic task involving semantically annotated resources. Various statistical data may easily be obtained such as information for the distribution of words in running text according to POS, rank–frequency for words (types), word senses, particular word senses.

Table 3 below shows the ten most frequent verb synsets, Table 4 gives an example of the distribution of the distinct senses of a word as annotated in BulSemCor - the verb *kazvam* (tell).

Rank	Frequency	Word	Definition
1	1075	sam:1	an auxiliary verb used in the formation of analytic verb forms
2	945	sam:9	be:4 have the quality of being; (copula used with adjective or a predicate noun)
3	226	sam:4	be:6 be identical to; be someone or something
4	182	tryabva:1, nalaga se:1, nalozhi se:1	must, have to
5	126	kazvam:2, kazha:2, izkazvam:1, izkazha:1, zayavyavam:1, zayavya:1	state:9, say:9, tell:7 express in words
6	118	sam:13, predstavlyavam:3, sastavlyavam:1, sastavyam:5	constitute:3, represent:13, make up:8, comprise:1, be:7 form or compose
7	116	imam:1, pritezhavam:1	have:12, have got:1, hold:29 have or possess, in a concrete or an abstract sense
8	102	moga:1	can, be able to, have the possibility or opportunity
9	100	moga:2	can, be able to, have the necessary means or skill or know-how or authority to do something
10	93	sashtestvuvam:3, sam:12, ima:5, bituvam:1	exist:1, be:3 have an existence, be extant

Table 3: Top 10 verb synsets

Freq	Sense	English sense/definition
31	kazvam:2; kazha:2; izkazvam:1; izkazha:1; zayavyavam:1; zayavya:1	state:9, say:9, tell:7 express in words
6	kazvam:5; :kazha5; saobshtavam:1; saobshtya:1	tell:4 let something be known
5	kazvam:4; kazha:4	say:10 state as one's opinion or judgment
2	naricham:1; nareka:1; kazvam:1; kazha:1; vikam:2; obrashtam se:2; obarna se:2	address:18, call:39 greet as with a prescribed form, title or name
2	glasya:1; pisha:7; kazvam:7;	read:12, say:12 have or contain a certain wording or form

Table 4: Frequency of selection of the available senses for *kazvam* (tell)

The numerical data along with the annotations provide empirical material for theoretical and applied lexicographic studies on sense distinctions and granularity, collocations, synonym selectional preferences (Table 5), lexical choices, largely applicable in dictionary compilation and wordnet development. It has recently been used in the study of Bulgarian idioms (Todorova 2010).

Frequency	Sense
39	chovek
19	lichnost
16	litse
5	individ
3	choveshko sashtestvo
1	dusha
1	smarten
0	osoba
0	persona

Table 5: Statistics for the selectional preferences for the literals of the synset *{person:1; individual:5; someone:1; somebody:1; mortal:5; human:4; soul:1}*

One of the major applications of the sense-annotated corpus is in corpus linguistics and computational linguistics, for instance for latent semantic analysis and in exploring the structure of the lexis. Above all BulSemCor is a valuable resource in word sense disambiguation tasks, where it may also serve as a gold standard for WSD annotation.

Future directions

The corpus is going to be made available to the research community and the wider audience on the web. To this end, a web-based interface is currently being developed.

Portions of the corpus are also to be employed for training and test data in experiments on training and testing WSD algorithms.

Acknowledgments

The financial support is granted under Contract No. BG051PO001–3.3.04/27 of 28 August 2009 within the Operation *Support to the development of PhD students, post-doctoral students, post-graduate students and young scientists* of the General Directorate *Structural Funds and International Educational Programmes* with the Ministry of Education, Youth and Science.

References

Fellbaum et al. 1998: Fellbaum, C. Performance and Confidence in a Semantic Annotation Task. In Fellbaum, C. (ed.) WordNet: An Electronic Lexical Database. Cambridge, Mass.: The MIT Press, pp. 217-237.

Filip 2008: Filip, H. Events and Maximalization: the Case of Telicity and Perfectivity. In Rothstein, S. (ed.) Theoretical and Crosslinguistic Approaches to the Semantics of Aspect. John Benjamins Publishing Company.

Koeva 1998: Koeva, S., Grammar dictionary of Bulgarian. Description of the Conception of the Language Data Organisation. – Bulgarian Language, 6, pp. 49-58.

Koeva et al. 2006: Koeva, S., Lesseva, S., Stoyanova, I., Tarpomanova, E., Todorova, M. Bulgarian Tagged Corpora. In Proceedings of the FASSBL-5 Conference, Sofia, 18-20 October 2006, pp. 78-86.

Koeva et al. 2008: Koeva, S., B. Rizov, S. Leseva. Chooser – a Multitask Annotation Tool. In Proceedings of the 6th Language Resources and Evaluation Conference, Marrakech, Morocco, 28-30 May, 2008.

Landes et al. 1998: Landes, Shari, Claudia Leacock and Randee Tengi. "Building Semantic Concordances". In Fellbaum, C. (ed.) Word-Net: An Electronic Lexical Database. Cambridge, Mass.: MIT Press, pp. 199-216.

Miller et al. 1993: Miller G. A., Leacock C., Randee T., and Bunker R. 1993. A Semantic Concordance. In Proceedings of the 3rd DARPA Workshop on Human Language Technology, pp. 303-308. Plainsboro, New Jersey.

Nitsolova 2008: Nitsolova, R. Balgarska gramatika. Morfologiya. Sofia.

Todorova 2010: Todorova, M. Organisation of the Linguistic Data in a Computational Morphological Dictionary of Bulgarian Verb Idioms. Proceedings of the Lexicography in the European Cultural Space Conference, Sofia, 2009, pp. 331-344.

(NOT SO) FREE WORD ORDER IN LAMBDA GRAMMAR: THE CASE OF BCMS[*,1]

Vedrana Mihalicek

Department of Linguistics, The Ohio State University
1712 Neil Ave, Columbus, OH 43210, USA
vedrana@ling.ohio-state.edu

ABSTRACT

While the word order in BCMS is in some respects quite flexible, in other respects it is fairly rigid. In this paper we try to show that Lambda Grammar (Muskens 2003, 2007), a variant of categorial grammar, is especially well suited for analyzing such word order patterns. This is because of a designated multimodal word order component which allows for fine-grained control over the linearization of constituents in a clause. We sketch the Lambda Grammar approach to word order with respect to the ordering of nouns, attributive adjectives and postnominal modifiers in BCMS.

1. Introduction

In this paper, we try to show that Lambda Grammar (Muskens 2003, 2007), a variant of categorial grammar, is extremely well suited for analyzing languages like Bosnian, Croatian, Montenegrin and Serbian (henceforth: BCMS) with (not so) free word order. While word order in BCMS is free in some respects (e.g. the order of phrasal constituents in a clause), it is fairly rigid in others (e.g. the relative order of nouns and post-nominal modifiers).

Lambda Grammar is particularly well suited for analyzing BCMS because of its rich word order component that allows for fine-grained control over linearization of expressions within phrases and larger utterances, without complicating the combinatorial syntactic component unnecessarily.

The remainder of the introduction gives an overview of Lambda Grammar, while the formal details of its word order component are presented in the appendix. Section 2 sketches a Lambda Grammar analysis of the ordering of attributive adjectives and postnominal modifiers, and left branch extraction in BCMS. In Section 3 we offer some concluding remarks.

1.1. Framework Overview

Lambda Grammar (Muskens 2003, 2007) is a kind of categorial (logic-based) grammar that contains three components: tectogrammar (combinatorial syntax), phenogrammar (word order)[2] and semantics.

We assume standard Montague-style semantics, with e and t as basic semantic types[3]. We use $\rightarrow$ as the only type constructor (analogous to Montague's $\langle$-,-$\rangle$).

The tectogrammatical component encodes syntactic dependencies between expressions. It consists of a set of linear types[4] corresponding to syntactic categories - for example, the type N corresponds to nouns, NP to noun phrases, and S to sentences. Intransitive verbs correspond to the type NP$\multimap$S, which intuitively reflects their need for a noun

[*] Many thanks to Carl Pollard for help with the more technical aspects of this work and for comments and suggestions on earlier drafts. The author would also like to thank Brian Joseph and Victor Friedman, as well as Marko Tadić and an anonymous reviewer for their invaluable help with selecting appropriate terminology.

[1] Here, BCMS abbreviates Bosnian, Croatian, Montenegrin and Serbian. We choose to use this term because we believe that the linguistic phenomena discussed here are common to all these languages. When differences are claimed to arise (for example, with respect to attributive adjectives) we will note which data pattern is attributed to which of the relevant languages.

[2] The terms 'phenogrammar' and 'tectogrammar' are due to Curry (1961) who programmatically advocates that the two components be formally distinguished.

[3] For expository simplicity, we use extensional semantics throughout.

[4] Tectogrammatical types are just formulas of the implicative fragment of linear logic (Girard 1987). The tectogrammatical signature is obtained by closing the set of basic types (N, NP, S...) under linear implication.

phrase subject argument. Similarly, transitive verbs correspond to the type NP⊸NP⊸S because they need and object and a subject noun phrase to form a sentence.

The tectogrammatical types, however, do not encode in any way the relative linear order of constituents. This is the responsibility of phenogrammar, the designated word order component. Following Oehrle (1994), typed lambda terms are used not only to represent meaning in the semantic component, but also word order in phenogrammar. There is only one basic phenogrammatical type **Str**, for 'string' and one type constructor → by means of which we can construct complex types such as **Str**→**Str**, or **Str**→**Str**→**Str**.

What allows for fine-grained control over word order is the presence of multiple operators in the phenogrammar. Intuitively, these operators glue expressions together in different ways, and how they are glued together determines the word order possibilities. The non-commutative operator ◦ (concatenation) disallows reordering of the constituents, and the commutative • allows permutation of constituents. We can therefore use these operators to control linearization. The formal details concerning the phenogrammatical logic and its interpretation are spelled out in the appendix.

The grammar consists of lexical entries (which are formally just non-logical *axioms*) and rules (formally, *inference rules* from the underlying logic). Each lexical entry consists of two typed lambda terms (for meaning and word order) and a tectogrammatical type. As a typographical convention, we bold the phenogrammatical constants, and append primes to semantic constants. We will present the rules as the need arises.

2. Data and Analysis

2.1. Free Word Order.

We start with the following lexicon for BCMS, to illustrate free word order of constituents in a clause[5]:

⊢**marija**: NomP: marija'[6]
⊢**anu**: AccP: ana'
⊢λpq.q•**vidi**•p: AccP⊸NomP⊸S: see'
⊢λq.q•**spava**: NomP⊸S: sleep'

Here, NomP corresponds to the category of nominative phrases, and AccP to accusative phrases[7]. Note that the phenogrammatical terms of the verbs make use of the commutative operator •. This allows the verb and its arguments to freely permute in the clause, as desired.

To combine the lexical entries above, we use the following rule, analogous to Merge in Minimalism:

$$\frac{\vdash \boldsymbol{a}:T\multimap U:a' \qquad \Gamma\vdash \boldsymbol{b}:T:b'}{\Delta,\Gamma\vdash \boldsymbol{a}(\boldsymbol{b}):U:\ a'(b')}\ [\multimap E]^{8} \quad \text{(cf. Merge)}$$

This is the rule of basic syntactic combination, allowing some expression of tectogrammatical type T⊸U to form a phrase of type U (its *result type*), once it combines with some expression of type T (its *argument type*). In

[5] For ease of exposition and since our goal is to illustrate the workings of the phenogrammatical component, we ignore gender and number, and pay minimal attention to case.

[6] The notation we use for lexical entries is slightly but inessentially different from Muskens (2003, 2007).

[7] We can treat NomP and AccP as subtypes of type NP, by interpreting the tectogrammatical signature into a preordered algebra; similarly for case-distinguished noun types. We suppress the details here due to space limitations.

[8] This rule corresponds to pointwise application in Muskens 2003, 2007.

phenogrammar and semantics, [⇨E] corresponds to function application. For example, given the lexicon above, we can construct a proof of the sentence *Marija spava*:

(1) $\underline{\vdash \lambda q.q \bullet \textbf{spava}: \text{NomP} \multimap \text{S}: \text{sleep}' \qquad \vdash \textbf{marija}: \text{NomP}: \text{marija}'}\,[\multimap E]$
$\vdash \textbf{marija} \bullet \textbf{spava}: \text{S}: \text{sleep}'(\text{marija}')$

Via lambda conversion in the phenogrammatical logic, the constant **marija** winds up to the left of **spava** once the two expressions combine, which is consistent with the intuition that the underlying word order in BCMS is SVO (Progovac 2005 argues that this is the case for Serbian). However, since **marija** and **spava** are combined via •, the two terms can permute, so the following is also a theorem of our grammar:

(2) $\vdash \textbf{spava} \bullet \textbf{marija}: \text{S}: \text{sleep}'(\text{marija}')$

Similarly, we can prove that *Marija vidi Anu* is a sentence:

(3) $\vdash \textbf{marija} \bullet \textbf{vidi} \bullet \textbf{anu}: \text{S}: \text{see}'(\text{ana}')(\text{marija}')$

Since all three phenogrammatical constants above are combined via the commutative operator, the other five possible orders of those three expressions in a clause are predicted to be possible, just as desired.

2.2. Post-nominal modification

Post-nominal modifiers in BCMS in general have to occur immediately to the right of the noun they modify. We illustrate with a post-nominal prepositional phrase:

(4) a. Marko voli djevojku iz Amerike.
marko-NOM loves girl-ACC from america-GEN
b. *Marko voli iz Amerike djevojku[9].
c. *Marko iz Amerike voli djevojku.
etc.

The entire noun phrase that contains the postnominal modifier can freely order with respect to other clausal constituents:

(5) a. Djevojku iz Amerike Marko voli.
b. Marko djevojku iz Amerike voli.
etc.

Further, the preposition *za* 'from', must occur immediately to the left of its genitive complement.

We easily account for this data by using the non-commutative operator in the phenogrammatical representation of the preposition. We add the following lexical entries:

$\vdash \lambda pq.q \circ (\textbf{iz} \circ p): \text{GenP} \multimap \text{Acc} \multimap \text{Acc}: \lambda x \lambda P \lambda y.\text{from}'(x)(y) \wedge P(y)$
$\vdash \textbf{djevojku}: \text{Acc}: \text{girl}'$
$\vdash \textbf{amerike}: \text{GenP}: \text{america}'$

Now, via [⇨E], our grammar generates *djevojku iz Amerike* in exactly that order, and prevents any permutations within that string:

(6) $\vdash \textbf{djevojku} \circ \textbf{iz} \circ \textbf{amerike}: \text{Acc}: \lambda y.\ \text{from}'(\text{america}')(y) \wedge \text{girl}'(y)$

[9] (4b) and (4c) are ungrammatical on the intended reading, where the prepositional phrase modifies djevojku 'girl'.

Permutation within the phrase is disallowed because the three constants, **djevojku**, **iz** and **amerike** are combined via the non-commutative operator ◦. However, the entire phrase can permute with other constituents in the clause because the linearization of clausal constituents is determined by the verb's lexical entry [10].

2.3. Attributive adjectives

We briefly illustrate how to analyze adjectival modification. We aim to show how the framework presented here could account for several distinct judgment patterns. We defer remarks about left branch extraction till the next section.

First we consider the most permissive dialect (e.g. this author's[11]) in which attributive adjectives can freely order with respect to other clausal constituents, as in:

(7)	a.	Ana	kupuje	novi	auto.
		ana-NOM	buys	new-ACC	car-ACC
		'Ana is buying a new car'			
	b.	Ana novi kupuje auto.			
	c.	Novi Ana kupuje auto.			
	d.	Novi Ana auto kupuje.			
	d.	Auto Ana novi kupuje.			
	e.	Auto Ana kupuje novi.			
	etc.				

To account for these judgments, we simply add the following lexical entry:

$\vdash \lambda p.$**novi** • p: Acc⇨Acc: $\lambda P \lambda x.P(x) \wedge$ new'(x)

Now, this adjective can freely permute with other clausal constituents, and all logically possible 24 orders of the four expressions in (7a) will be generated by the grammar.

To account for the pattern of judgments in which the attributive adjective must occur immediately to the left of the noun it modifies, but the entire resulting phrase can permute with respect to other clausal constituents (e.g. Leko 1999 in his paper about Bosnian noun phrases claims that adjectives cannot follow nouns unless they are coordinated or themselves modified), we slightly modify the lexical entry given above:

$\vdash \lambda p.$**novi** ◦ p: Acc⇨Acc: $\lambda P \lambda x.P(x) \wedge$ new'(x)

With this lexical entry, the grammar will predict that 6 orders of the four expressions in (7a) are possible - exactly those in which the adjective occurs immediately to the left of the noun. The permutation of the adjective and the noun is impossible, however, because of the non-commutative operator in the adjective's lexical entry and lack of any mixed associativity between the two operators.

Finally, we account for an intermediate judgment pattern, where the noun and the adjective can permute, but must remain contiguous in the clause (see e.g. Zlatić 1997 who recognizes that in Serbian noun phrases even ordinary attributive adjectives with no complements can follow the noun they modify). We retain the lexical entry for the adjectives which combines with the noun in the commutative way, but we systematically alter the lexical entries for verbs (and other expressions that take nominal complements) as follows:

---

[10] In order to promote this expression to the status of the full noun phrase (AccP) instead of just a noun with modifiers (Acc) we can introduce a rule that would convert it into an existentially quantified expression of the appropriate (phrasal) tectogrammatical type. We omit the presentation of that rule here due to space constraints and since our focus is on illustrating the workings of the phenogrammar.

[11] The author was born and raised in Sarajevo, Bosnia and Herzegovina, and lived for many years in Zagreb, Croatia.

$\vdash \lambda pq.\Diamond_y q \bullet \mathbf{kupuje} \bullet \Diamond_y p$: AccP$\multimap$NomP$\multimap$S: buy'

$\vdash \lambda q.\Diamond_y q \bullet \mathbf{spava}$: NomP$\multimap$S: sleep'

$\Diamond_y$ introduced in these lexical entries is a unary operation on strings, that intuitively creates a word order 'fortress'. It does not affect combinatorics within the string it combines with, but it prevents any substrings from 'escaping' it. In the relevant case, while the noun and the adjective can permute, $\Diamond_y$ will prevent both of them from scrambling outside of the noun phrase and into the larger clausal environment. With these lexical entries, the grammar will predict that exactly 12 orders of the four expressions in (7a) are possible - all those in which the adjective and the noun are contiguous.

2.4. Left Branch Extraction

Consider *wh* questions such as the following:

(8)	a.	Koji	auto	Marija	kupuje?
		which-ACC	car-ACC	marija-NOM	buys
		'Which car is Marija buying?'			
	b.	Koji Marija auto kupuje?			
	c.	Koji Marija kupuje auto?			
	etc.				

While the entire *wh* phrase *koji auto* 'which car' semantically scopes over the clause, it is possible to only extract (i.e. move to the left periphery) the interrogative determiner *koji* 'which'. We straightforwardly account for this instance of so-called left branch extraction by giving the following lexical entry for *koji*:

$\vdash \lambda p \lambda f.\mathbf{koji} \circ p \bullet f(e)$: Acc$\multimap$(AccP$\multimap$S)$\multimap$Q: which'[12]

The interrogative determiner first combines with a noun (Acc), then with a sentence with a bound accusative trace (AccP$\multimap$S), to yield a question (Q). The phenogrammatical term of *koji* ensures that it occurs on the left periphery of the question, but it allows its complement noun to freely permute with the expressions in the sentence with a bound trace. To illustrate how this works, we work through one example in detail. For that, we need the other two logical rules that our grammar contains:

$$\frac{}{p{:}T{:}x \vdash p{:}T{:}x}[\mathrm{Ax}] \qquad \text{(cf. Trace)}$$

This rule lets us introduce traces (hypotheses) into derivations, that are then kept track of in the context (to the left of the turnstile $\vdash$). *p* (*x*) is a metavariable over phenogrammatical (semantic) variables. We bind the trace (by abstracting on the free variables in the phenogrammatical and semantic term) by means of this rule, analogous to Move in Minimalism:

$$\frac{\Delta, p{:}T{:}x' \vdash \mathbf{b}{:}U{:}r'}{\Delta \vdash \lambda p.\mathbf{b}{:}T\multimap U{:}\ \lambda x.r'}[\multimap \mathrm{I}]^{13} \qquad \text{(cf. Move)}$$

[12] Here, the semantic constant which' abbreviates $\lambda P \lambda Q \lambda x \lambda p.[(p \wedge P(x)) \wedge (p=Q(x) \vee p=\neg Q(x))]$. That is, we analyze constituent questions meanings as functions from individuals to a singleton set of propositions. Intuitively, the meaning of a question is analyzed as a set of possible answers to it, in the spirit of Karttunen 1977.

[13] This rule corresponds to pointwise abstraction in Muskens 2003, 2007.

Now we assemble the question in (8a). First we introduce the object trace and proceed to combine it with the verb. We combine the verb phrase (still containing the trace) with the subject noun phrase. Finally, we bind that trace by means of [⇾I].

(9)

$$\dfrac{\dfrac{\vdash \lambda pq.q \bullet \textbf{kupuje} \bullet p\text{: AccP}\multimap\text{NomP}\multimap\text{S: buy'} \quad a\text{:AccP:}x \vdash a\text{:AccP:}x\ [\multimap E] \qquad \vdash \textbf{marija}\text{: NomP: marija'}}{a\text{:AccP:}x \vdash \lambda q.q \bullet \textbf{kupuje} \bullet a\text{: NomP}\multimap\text{S: buy'}(x)}\ [\multimap E]}{a\text{:AccP:}x \vdash \textbf{marija} \bullet \textbf{kupuje} \bullet a\text{: S: buy'}(x)(\text{marija'})}\ [\multimap I]$$

$$\vdash \lambda a.\ \textbf{marija} \bullet \textbf{kupuje} \bullet a\text{: AccP}\multimap\text{S: } \lambda x.\text{buy'}(x)(\text{marija'})$$

The result of this derivation (i.e. the conclusion of this proof) is a sentence with a bound accusative trace. Next we assemble the *wh* phrase *koji auto* 'which car':

(10)

$$\dfrac{\vdash \lambda p \lambda f.\textbf{koji} \circ p \bullet f(e)\text{: Acc}\multimap(\text{AccP}\multimap\text{S})\multimap\text{Q: which'} \qquad \vdash \textbf{auto}\text{:Acc:car'}}{\vdash \lambda f.\textbf{koji} \circ \textbf{auto} \bullet f(e)\text{:(AccP}\multimap\text{S)}\multimap\text{Q: which'(car')}}\ [\multimap E]$$

Now we can combine this *wh* phrase with the sentence with a bound trace. Note that the phenogrammatical term of *Marija kupuje* contains a bound variable (a), intuitively the accusative gap, so its type is **Str→Str**. *koji auto* will plug this gap, by feeding its argument the null string e (the two-sided identity for the binary phenogrammatical operators), thereby turning it into a term of type **Str**. Combining the conclusions of the derivations in (9) and (10) we get:

$\vdash$**koji** ◦ **auto** • **marija** • **kupuje**:Q: which'(car')(λx.buy'(x)(marija'))

We can derive as theorems any permutation of *auto*, *marija* and *kupuje*. However, *koji* must occur on the left periphery and cannot mix in with the rest of the constituents, since we didn't admit any mixed associativity between the two binary phenogrammatical operators. This analysis could in principle be extended to examples of left-branch extraction of non-interrogative determiners and adjectives.

3. Conclusion

We tried to show that a framework like Lambda Grammar, because of its rich, multimodal word order component, is very suitable for an analysis of languages like BCMS which have (not so) free word order. This was illustrated with respect to the order of clausal constituents, post-nominal modifiers, several sets of judgments concerning attributive adjectives and, finally, extraction of *wh* determiners.

The framework is completely formalized, and, in our opinion, simple and straightforward. Word order complexities were dealt with in the phenogrammar, without unnecessary complications in the combinatorial syntactic component. Although simple, the formalism is, as we tried to show, expressive enough to be used for analyses of complex linguistic phenomena.

We completely ignored pragmatics and prosody in this paper. It is our hope that eventually the analyses sketched here (and the framework itself) can be elaborated to include that kind of information as well. We also hope to extend the analysis presented here to enclitics and multiple *wh* questions.

References

Curry, Haskell. 1961. Some logical aspects of grammatical structure. In Jakobson, Roman (ed.), *Structure of Language and Its Mathematical Aspects.*

Girard, Jean-Yves. 1987. Linear logic. In *Theoretical Computer Science:* 50.

Karttunen, Lauri. 1977. Syntax and semantics of questions. In *Linguistics and Philosophy*: 1(1).

Leko, Nedžad. 1999. Functional categories and the structure of DP in Bosnian. In Dimitrova-Vulchanova, M. and L. Hellan, eds., *Topics in South Slavic Syntax and Semantics*. John Benjamins.

Muskens, Reinhard. 2003. Language, lambdas, and logic. In Kruijff, G. and R. Oehrle, eds., *Resource Sensitivity in Binding and Anaphora*. Kluwer.

Muskens, Reinhard. 2007. Separating syntax and combinatorics in categorial grammar. *Research on Language and Computation*: 3(5)

Oehrle, Richard. 1994. Term-labeled categorial type systems. In *Linguistics and Philosophy*: 17(6).

Progovac, Ljiljana. 2005. *A Syntax of Serbian Clausal Architecture*. Bloomington: Slavica.

Zlatić, Larisa. 1997. *The Structure of the Serbian Noun Phrase.* PhD thesis, University of Texas at Austin.

Appendix. Phenogrammar - Formal Details.

The multimodal phenogrammatical component we used is based on Muskens 2003, 2007. Here we present the essential formal details.

The interpretation of types. The phenogrammatical type **Str** is an abbreviation for the type String→Bool. This is a technical choice given that we model phenogrammar in a Kripke frame. Call the set that interprets the type String S. The phenogrammatical constants are then interpreted as subsets of S, so their type is String→Bool.

The interpretation of operators. We define a ternary relation R° that interprets ◦ for t, t'⊆S and k,k',k''∈S as λtt'λk.∃k'k''[R°kk'k'' ∧ k'∈t' ∧ k''∈t'']. We impose suitable frame conditions to guarantee that ◦ behaves like concatenation:

$\forall kk'k_1k_2[(R^{\circ}kk_1k_2 \wedge R^{\circ}k'k_1k_2) \Rightarrow k=k']$ [uniqueness]

$\forall kk_1k_2k_3[\exists k'(R^{\circ}kk'k_3 \wedge R^{\circ}k'k_1k_2) \Leftrightarrow \exists k'(R^{\circ}kk_1k' \wedge R^{\circ}k'k_2k_3)]$ [associativity]

$\forall kk'[R^{\circ}kk'1 \Rightarrow k=k']$ [1 is the right identity for ◦]

$\forall kk'[R^{\circ}k1k' \Rightarrow k=k']$ [1 is the left identity for ◦]

The singleton set containing 1 is the denotation of the phenogrammatical constant e. So, in the phenogrammatical logic, e behaves like the two-sided identity for the binary operations.

We analogously define a ternary relation $R^{\bullet}$ that interprets •. We impose the same frame conditions on $R^{\bullet}$ as on R° and add the following one to ensure that • is commutative:

$\forall kk'k_1k_2[R^{\bullet}kk_1k_2 \Leftrightarrow R^{\bullet}k'k_2k_1]$ [commutativity]

The two binary operators are related in the model via the following frame condition:

$\forall kk_1k_2\ [R^{\bullet}kk_1k_2 \Rightarrow R^{\circ}kk_1k_2]$

Crucially, we do not admit any other interaction postulates for these two operators (such as mixed associativity).

$\Diamond_y$ is formally just a unary modality. y is mnemonic for 'yield' We define a binary relation on S that interprets $\Diamond_y$ as λtλk.∃k'[R^{y}kk' ∧ k∈t]. We add the following frame conditions:

$\forall kk'[(\forall k_1k_2(R^{\circ}kk_1k_2 \Rightarrow (k_1=1 \vee k_2=1)) \wedge R^{y}k'k) \Rightarrow k=k']$

$\forall kk_1k_2[(R^{y}kk_1k_2 \wedge R^{y}kk_2k) \Rightarrow k_1=k_2]$

$\forall kk_1k_2k_3[(R^{y}kk_1 \wedge R^{\bullet}k_1k_2k_3) \Rightarrow \exists k_4k_5(R^{\circ}kk_4k_5 \wedge R^{y}k_4k_2 \wedge R^{y}k_5k_3)]$

Inference rules in the phenogrammatical logic. The model supports the following inference rules in the phenogrammatical logic.

$$\frac{}{A \bullet B \sqsubseteq B \bullet A}\ [\text{COMM}] \qquad \frac{}{B \bullet A \sqsubseteq A \bullet B}\ [\text{COMM}]$$

$$\frac{}{(A \times B) \times C \sqsubseteq A \times (B \times C)}\,[ASS] \qquad \frac{}{(A \times B) \times C \sqsubseteq A \times (B \times C)}\,[ASS] \quad \text{where } x=\{\bullet, \circ\}$$

$$\frac{A \sqsubseteq A'}{A \times B \sqsubseteq A' \times B}\,[MON] \qquad \frac{A \sqsubseteq A'}{B \times A \sqsubseteq B \times A'}\,[MON] \quad \text{where } x=\{\bullet, \circ\}$$

$$\frac{A \sqsubseteq B \qquad B \sqsubseteq C}{A \sqsubseteq C}\,[TRANS]$$

$$\frac{}{A \times e \sqsubseteq A}\,[ID] \qquad \frac{}{e \times A \sqsubseteq A}\,[ID] \quad \text{where } x=\{\bullet, \circ\}$$

$$\frac{}{A \bullet B \sqsubseteq A \circ B}\,[PRON]$$

$$\frac{}{\Diamond_y (A \times B) \sqsubseteq \Diamond_y A \circ \Diamond_y B}\,[Y1] \quad \text{where } x=\{\bullet, \circ\}$$

$$\frac{}{\Diamond_y A \sqsubseteq A}\,[Y2] \quad \text{if A is a lexical expression or A=e}$$

$$\frac{}{\Diamond_y \Diamond_y A \sqsubseteq \Diamond_y A}\,[Y3]$$

Interface rule. This rule allows us to replace some phenogramatical term **a** with a phenogrammatical term **b** if **a**⊑**b** in the phenogrammatical logic, without changing the tectogrammatical type or the meaning of the expression in question.

$$\frac{\Delta \vdash \mathbf{a} : T : r'}{\Delta \vdash \mathbf{b} : T : r'}\,[\sqsubseteq] \quad \text{if } \mathbf{a} \sqsubseteq \mathbf{b}$$

ROBUST KEYPHRASE EXTRACTION FOR A LARGE-SCALE CROATIAN NEWS PRODUCTION SYSTEM

Jure Mijić, Bojana Dalbelo Bašić, Jan Šnajder

University of Zagreb, Faculty of Electrical Engineering and Computing
Unska 3, Zagreb, Croatia
{jure.mijic, bojana.dalbelo, jan.snajder}@fer.hr

ABSTRACT

Summarizing an article with just a few keyphrases can be a difficult task, even for trained experts. Large-scale keyphrase extraction requires a method that is fast and reliable, and yet relatively effective. In this paper we describe such a keyphrase extraction system developed for a large-scale Croatian news production system. We describe how the system works and evaluate the implemented keyphrase extraction methods using a gold set annotated by human annotators. The results indicate that, despite the simplicity of our approach, the performance of the system is comparable to that of the human annotators.

1. Introduction

Enrichment of documents with metadata can be done in many ways, one of which is the addition of extracted keyphrases from the text of the document. The extracted keyphrases are expected to represent the most relevant information contained in the document. In *keyword assignment*, the keyphrases are chosen from a predefined taxonomy. Assigned keyphrases are often very general and thus mostly used to describe general information, which is why they are often treated as categories. Unlike keyphrase assignment, *keyword extraction* enriches a document with keyphrases that are explicitly mentioned in the text. The advantages of keyphrase extraction over keyphrase assignment are that the method is not limited to a predefined set of keyphrases and can better link documents across category domains. While keyphrase assignment can be used to organize documents in a hierarchical structure of categories or keyphrases, extracting keyphrases from text enables the enrichment of the document with much more specific metadata.

Many methods for keyphrase extraction have been developed for the English language. Commonly used machine learning methods for keyphrase extraction are supervised methods such as the C4.5 decision tree algorithm (Turney 2002) and the more popular Bayes classifier used in the KEA system (Witten et al. 1999), later improved to KEA++ (Medelyan and Witten 2006). With the exception of work by Ahel et al. (2009), there is no published research on keyword extraction for Croatian. Ahel et al. (2009) describe a system based on a Bayes classifier and a candidate generation method similar to the one used in the KEA system.

In the news domain, the articles commonly convey very specific information, e.g., about specific events in time, and they are often topically related to other articles from different points in time. Using a predefined taxonomy to annotate every different event or topic in time would require a large number of keyphrases and a constant addition of new keyphrases for emerging events and topics. The main purpose of the extracted keyphrases is to enable horizontal, cross-category linkage between related documents, e.g., to extract specific keyphrases often used by journalists in the description of various affairs and events.

This paper describes a keyphrase extraction system that is used in a real-world news production system that processes a large number of documents, measured in tens of thousands, on a daily basis. The news production system is operating in one of the largest Croatian news agencies, and the documents processed are news articles written in Croatian language. Our aim was to implement a simple yet effective keyphrase extraction method that would perform fast and reliable, and would require as little maintenance as possible. In the design of our keyphrase extraction system, tradeoffs were made in favor of speed, reliability, and low maintenance requirements over qualitative performance.

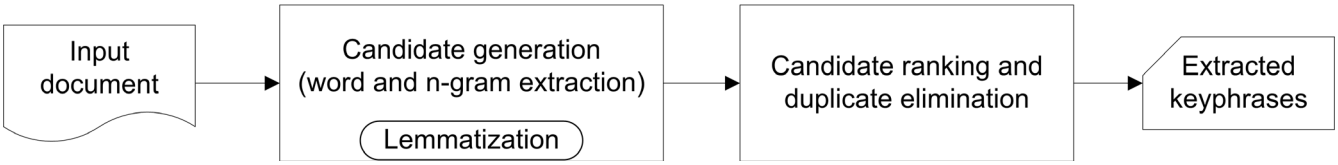


Figure 1: The keyphrase extraction process

The rest of the paper is organized as follows. In the following section we describe how our keyphrase extraction system works and discuss some of its particularities. In Section 3 we describe the evaluation methodology and discuss the experimental results. Section 4 concludes the paper.

2. Robust keyphrase extraction system

This section gives an overview of our keyphrase extraction system, along with some tradeoffs that we made when designing the system. We highlight some of the additions to the system that were necessary to ensure the reliability of continuous operation within a large-scale news production system.

2.1. Performance vs. complexity

Implementing a fast, reliable, and effective system that will be used in real-world applications almost always implies that some tradeoffs have to be made in the system's design. Choosing a more complex method might yield better qualitative performance, but at a cost of lower speed and – in many cases – lower reliability. Operation inside a large-scale system requires that the integrated modules operate with low response time. The modules are also used in a multiprocess environment, therefore adequate multithreaded techniques must be implemented.

In our case the tradeoff was made by implementing a simpler unsupervised method which ensures good performance and requires little to no maintenance, instead of implementing a more complex method based on supervised machine learning methods. A supervised method could yield slightly better keyphrase extraction quality, but would require more computational resources and more maintenance. Moreover, supervised methods would require the system to be periodically retrained. This is in contrast with the unsupervised method that we have implemented, which relies only on statistical data gathered from the text that the system has already processed. That statistical data is automatically updated during the operation of the system and no human intervention is required.

2.2. Extraction system architecture

The keyphrase extraction system is designed to operate autonomously, meaning that it only requires an initial setup and no further intervention is needed. The core method used in the keyphrase extraction is unsupervised, therefore it requires no prior annotation of data. The method is based on the statistical data such as keyphrase and document frequency, and can extract keyphrases consisting of up to four words. The initial setup consists of defining the maximum number of statistical data that the system holds, and the initial collection of statistical data.

The keyphrase extraction process is illustrated in Fig. 1. The first step is the candidate generation, in which all possible keyphrase candidates are extracted from the text of the document, along with their respective frequencies in the document. In this step, lemmatization is used to for the morphological processing of the keyphrase candidates. In the second step, the candidates are ranked based on statistical data. After the candidates are ranked, the system returns a user-defined number of top-ranked keyphrases.

The keyphrase extraction system retains the statistical data for every document processed so far, i.e., the document frequencies for all words and all n-grams. In order to avoid that the same document is processed

multiple times, each document text is hashed and the hash value is stored in the system.

Although our keyphrase extraction method can potentially extract named entities as keyphrases, we chose not to extract keyphrases containing named entities because named entity recognition is handled by another module in the news production system (Bekavac and Tadić 2007).

2.3. Candidate generation

We have experimented with three approaches to candidate generation: single-word candidates, POS-filtered keyphrase candidates, and MSD-filtered keyphrase candidates. In all three approaches, the words are first lemmatized with the morphological lexicon described by Šnajder et al. (2008) to eliminate the effect of morphological variation.

Single-word candidates are obtained by extracting from the document all words except the stop words from a predefined list. The list of stop words contains the so-called functional words (conjunctions, prepositions, interjections, pronouns, particles and also numbers), which cannot constitute a keyphrase on their own.

To extract the POS-filtered keyphrase candidates, we first extract from the document all words and all n-grams of up to length four, except for those crossing the sentence boundaries (which we determine with regular expression matching). We then use the lemmatizer to (ambiguously) tag the words and the n-grams. Based on the obtained POS-patterns, we filter out the candidates that do not qualify as valid keyphrases: those which contain a verb, begin with a stop word, or end with a stop word.

The MSD-filtered keyphrase candidates are obtained similarly as the POS-filtered candidates, except that MSD (morphosyntactic description) filters also encode the morphological categories of case, number, and gender. The filtering is therefore more precise because it takes into account the syntactic relationships between the words constituting a keyphrase. For (ambiguous) MSD tagging we use the same lemmatizer as above. To create the MSD patterns, we use a small (15 rules in total) unification-based grammar. Using this grammar in a generative fashion, we created a set of 6,750 MSD keyphrase patterns.

Note that the candidate generation procedure is the only language-dependent processing within our system. The next step – candidate ranking – is language independent and uses only the statistical information extracted from text.

2.4. Candidate ranking

Ranking of keyphrase candidates is done using the following formula:

$$score(t_i) = tf_i \cdot idf_i \tag{1}$$

where tf_i is the normalized frequency of the keyphrase candidate in the document being processed, and idf_i is the inverse document frequency of the keyphrase. Normalized keyphrase frequency is calculated as:

$$tf_i = \frac{n_{i,j}}{\sum_k n_{k,j}} \tag{2}$$

where $n_{i,j}$ denotes the number of times the keyphrase occurs in the document d_j, and the denominator denotes the sum of number of occurrences of all keyphrases in document d_j. The inverse document frequency is calculated as follows:

$$idf_i = \log \frac{|D|}{|\{d : t_i \in d\}|} \tag{3}$$

where $|D|$ denotes the total number of documents in the statistical data stored in the keyphrase extraction system, and $|\{d : t_i \in d\}|$ denotes the number of documents in which the term t_i appears, i.e., $n_{i,j} \neq 0$.

The candidate generation process may generate overlapping keyphrase candidates, i.e., pairs of n-grams for which one n-gram subsumes the other. In order to avoid extracting similar keyphrases, after the candidates have been generated, the system eliminates those that overlap. The choice of which of the overlapping keyphrase candidate is most appropriate is left to the TF-IDF ranking procedure: keyphrases are ranked by their TF-IDF score and keyphrases that overlap with another higher-ranked keyphrase are discarded.

2.5. System self-maintenance

An important feature of the implemented keyphrase extraction method is that is requires no maintenance. In other words, the method is self-maintaining in that the overall effectiveness of the method does not deteriorate over time. This is important because the keyphrase extraction system is meant for continuous operation in a real-world news production system and supposed to be running for a long period of time. A problem that arises when using the statistical data (such as document frequency) is that frequency of the keyphrases varies over time and also the emergence of new keyphrases. To address this, our method automatically updates the keyphrase vocabulary and all statistical data over time by retaining the relevant data only for the most recent set of distinct documents. The documents are identified by hashing the text in the body of the document. Maximum number of documents for which the system holds the data is defined in the initial setup of the system.

The storage for statistical data in our system is partitioned into a preconfigured number of blocks. The blocks are filled one by one with statistical data from newly processed documents. The hash for each processed document is also stored in the same block as the statistical data for that document. When a block is filled with a maximum number of documents, the storage of statistical data is shifted to the next empty block. If all the blocks are full, data from the oldest block is deleted and that block is freed to be used to accumulate the statistical data for new documents.

Given the nature of operation of a news production system – the continuous throughput of the most recent news articles – the keyphrase extraction method will always hold the statistical data related to the most recent use of keyphrases in document processed by the production system. Thus, the newly emerged keyphrases will not be ignored by the extraction method, but rather they will have a relevance higher than other more common keyphrases with similar frequencies. The extraction method can be configured to hold statistical data for any number of documents, and it does not require a large number of documents for the initial setup (the initial collection of statistical data). After the system is configured and is put into operation, it automatically takes care of the updates and the persistence of the statistical data, making any further maintenance unnecessary.

3. Evaluation

We aimed at performing a thorough evaluation of our keyphrase extraction method. A problem one must face when evaluating keyphrase extraction is that this is a highly subjective task. Even human annotators are having difficulty to agree upon the types of keyphrases that should be extracted. For example, should more general keyphrase candidates have precedence over the more specific ones or vice versa? The evaluation method for keyphrase extraction should therefore consider a number of possible relations between the extracted and the expected keyphrases. An overly general keyphrase is still better than a totally incorrect one, and morphological variations of keyphrases shall also be taken into account. To enable an objective and conclusive evaluation of our keyphrase extraction method, we have performed the evaluation as follows.

Table 1: Inter-annotator agreement calculated in terms of the F_2 measure

Annotator	1	2	3	4	5	6	7	8
1	–	0.521	0.516	0.503	0.457	0.501	0.382	0.339
2	0.457	–	**0.523**	**0.565**	0.497	0.540	0.340	0.304
3	0.517	**0.598**	–	**0.661**	0.507	0.576	0.404	0.407
4	0.433	**0.555**	**0.568**	–	0.490	0.592	0.339	0.293
5	0.398	0.495	0.440	0.495	–	0.533	0.391	0.315
6	0.413	0.508	0.474	0.566	0.505	–	0.361	0.316
7	0.438	0.444	0.462	0.450	0.514	0.500	–	0.415
8	0.418	0.426	0.501	0.417	0.444	0.469	0.447	–

3.1. Gold set

We have developed a gold set composed of Croatian news articles. To the best of our knowledge, no such set for Croatian language has yet been made publicly available. The set contains 1020 news articles, randomly chosen from a set of 40,000 articles provided by the Croatian News Agency (HINA). Two basic criteria were used in the selection of the news articles: the minimum and the maximum size of the document. The minimum size of the document was set to one hundred words and the maximum to one million words.

The documents were annotated by eight human expert annotators. Detailed annotating instructions were compiled and handed out to the annotators. The annotators agreed upon a set of conventions for keyword extraction, but annotated the documents independently. A common subset of 60 documents was assigned to all eight annotators for the purpose of calculating the inter-annotator agreement. The remaining 960 articles were divided in subsets of 120 articles and annotated separately by each of the eight annotators. The overlapping 60 documents are used for the evaluation in this paper; the non-overlapping 960 documents are reserved to be used for supervised learning methods.

We selected keyphrases extracted by three of the annotators as the gold standard set. We chose the three annotators based on inter-annotator agreement measured in terms of the F_2 measure, which is the measure we seek to optimize (cf. Section 3.2.). The choice was done based on the maximum average inter-annotator agreement for all combinations of three out of eight annotators. The inter-annotator agreement values are given in Table 1. The remaining five annotators were scored against the gold standard set, and the average human annotator performance was calculated.

3.2. Evaluation methodology

Because of the nature of keyphrase extraction, the evaluation method for keyphrase extraction is somewhat different from standard methods used in classification. The matching of the extracted keyphrases to the gold standard keyphrases must include some kind of approximate matching. One cause of difference between the extracted keyphrases and gold standard keyphrases is morphological variation. Morphological variants of the extracted keyphrases should also be considered as true positive matches, especially if the system that uses the extracted keyphrases can handle such morphological variations (as is the case in the targeted news production system). Another type of approximate matching is matching of more specific (i.e., longer) keyphrases to the more general (i.e., shorter) keyphrases, and vice versa. This kind of approximate matching should also be included because, even if the extracted keyphrase more specific or more general than the gold standard keyphrase, it still retains a part of the relevant information.

To address the above issues, we have used an approximate matching method proposed by Zesch and Gurevych (2009), with a few modifications. They define four types of keyphrase matches: *Exact*, *Morph*,

Includes, and *PartOf*. The *Exact* match is a straightforward match where the extracted keyphrase and the gold standard keyphrase are identical. The *Morph* match accounts for the morphological variations of words in the keyphrase. The *Includes* is a match in which the extracted keyphrase includes the gold standard keyphrase, whereas *PartOf* is as a match in which the extracted keyphrase is a part of the gold standard keyphrase. We extended the described types of approximate matching with morphological variations of the *Includes* and the *PartOf* match, i.e., the *IncludesMorph* and the *PartOfMorph* types of matches.

The matching of the extracted keyphrases to the gold standard keyphrases includes a mechanism of keyphrase exclusion, i.e., the matched gold standard keyphrase cannot be matched again to any other extracted keyphrase. That way any similar or duplicate extracted keyphrases is penalized. With the introduction of the described approximate matches a specific problem arises. It is possible that an extracted keyphrase matches to multiple gold standard keyphrases, and by choosing one of the matches we limit the potential matches for other extracted keyphrases. Therefore the matching procedure must determine the optimal matching, i.e., the one in which the maximum number of extracted keyphrases can be matched.

Regarding the performance measure, we decided to put more emphasis on recall than on precision. Because the main purpose of the extracted keyphrases is cross-category linkage between documents, we felt that it is more important not to miss the potential links between the related documents (i.e., extract more keyphrases) than to potentially miss some of the links (i.e., extract fewer, but more precise keyphrases). For this reason, we aimed at optimizing the F_2 measure instead of the more commonly used F_1 measure. The F_2 measure gives two times as much importance to recall as precision, and is defined as $F_2 = (5PR)/(4P + R)$.

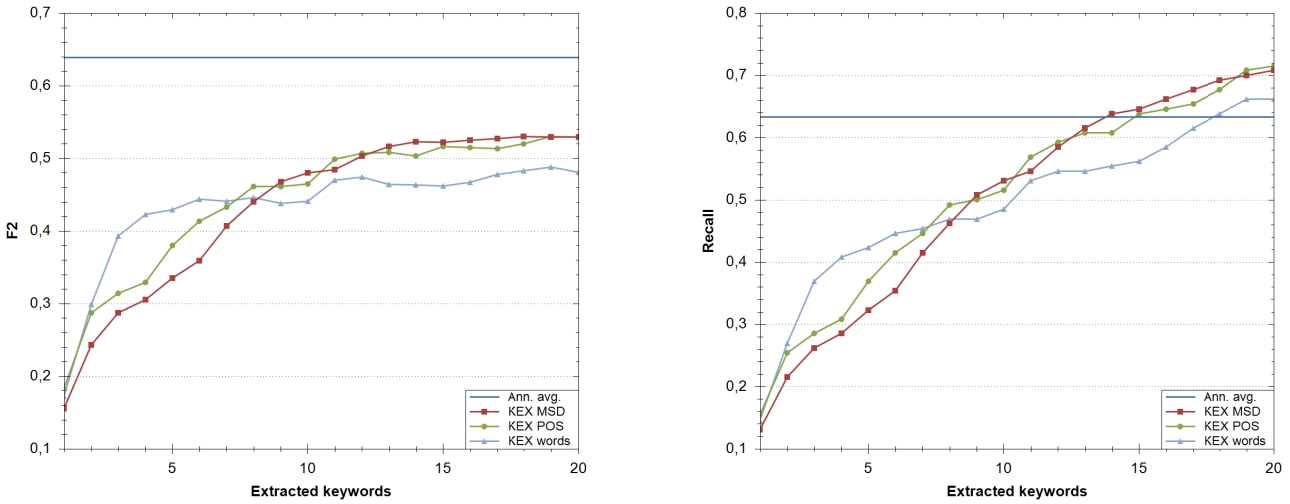
One further departure we made from the standard evaluation methodology is the way we computed the F_2 measure. In order to account for the high subjectivity of the keyphrase extraction task, which is evident from the low inter-annotator scores, we decided to calculate the F_2 measure asymmetrically as follows. We computed the precision with respect to the union of the results of the three gold standard annotators. Conversely, we computed the recall with respect to the intersection of the results of the tree gold standard annotators. This makes the F_2 measure tolerant to a false positive keyphrase that was assigned to a document by at least one of the three annotators. Conversely, the measure is tolerant to a false negative keyphrase that was not assigned to a document by all three annotators.

3.3. Evaluation results

We have analyzed three candidate generation methods in our keyphrase extraction system: n-gram extraction with MSD-filtering, n-gram extraction with POS-filtering, and extraction of single-word keyphrases (considered as the baseline). We have also compared the performance of our system to the average human annotator performance. Fig. 2 shows the performance results measured in terms of F_2 and recall with respect to the number of extracted keywords. The results reveal that extracting more than 15 keyphrases does not bring any improvement in terms of the F_2 measure. Extracting more keyphrases does improve recall, but significantly decreases the precision. Therefore, in the analysis that follows, we set our system to extract 15 keywords.

Our method achieved an F_2 score of 0.522, while the average F_2 score for human annotators was 0.639 and the lowest performing annotator scored an F_2 score of 0.51. If we analyze recall – the measure that we focused on – we observe that the difference in performance between our method and human annotators is even lower: the recall of our method is 0.646, while the recall for human annotators is 0.632 on average and 0.485 at worst. Extracting n-gram keyphrases proved to contribute to the quality of the extracted keyphrases. Keyphrase extraction with POS-filtering and MSD-filtering yielded similar performance. While extraction of single-word keywords yielded better performance for the top-ranked keywords, the extracted keywords are very general and most of the matches were *PartOf* or *PartOfMorph* matches.

Table 2 shows the types of matches for our three methods and the average number of types of matches for the five annotators whose annotations were not included in the gold set. Extracting only single-word



(a) (b)

Figure 2: The performance of keyword extraction compared against the average human performance, measured in terms of (a) F_2 measure and (b) recall.

keywords significantly lowers the number of *Exact* matches, scoring only 25 matches, as most of the gold set keyphrases contain more than one word. Extracting multiword keyphrases improves the number of *Exact* matches, scoring 40 matches for the MSD-filtered n-gram extraction and 47 for the POS-filtered n-gram extraction. N-gram extraction using MSD filters produces more specific (i.e., longer) keyphrases compared to keyphrases extracted using POS filters, which in turn results in a slightly lowered *Exact* matching and a higher *Includes* and *IncludesMorph* matching. Both n-gram extraction methods resulted in a similar number of *PartOf* and *PartOfMorph* matches. Average number of *Exact* matches for the five human annotators is significantly higher than for any of our methods, indicating the establishment of a well-defined keyphrase extraction convention among the human annotators. The fact that for human annotators the number of *Includes* and *IncludesMorph* matches is larger than the number of *PartOf* and *PartOfMorph* matches indicates that even human annotators have a tendency to extract more specific (i.e., longer) keyphrases.

We have also compared our method with the more complex supervised method based on the naïve Bayes classifier, developed by Ahel et al. (2009). As expected, this method yields slightly better results (F_2 score of 0.569). Incidentally, the feature that was proven to be the most useful for the naïve Bayes classifier is the TF-IDF score, which is also the feature used by our method.

The above results indicate that, although our method is unsupervised, its performance is comparable to that of the supervised methods and the human annotators.

4. Conclusion

We have developed a robust keyphrase extraction system that is designed for use in a large-scale Croatian news production system. We described the overall system architecture and pointed out some of its characteristics, such as self-maintenance and the reliability of the system. We implemented three keyphrase extraction methods: single-word keyphrase extraction (used as the baseline), POS-filtering, and MSD-filtering. The methods were tested on a set annotated by eight expert human annotators, and annotated keyphrases from three human annotators were used as the gold standard. In the evaluation we used an approximate matching technique that accounts for expected variations in the extracted keyphrases.

The results show that the extraction of multiword keyphrases significantly contribute to the system

Table 2: Number of matches for different types of matches and different candidate generation methods

	KEX MSD	KEX POS	KEX words	Annotator avg.
Exact	40	47	25	80
Morph	12	17	30	10
Includes	66	49	0	19
IncludesMorph	19	17	0	5
PartOf	6	6	23	9
PartOfMorph	7	11	51	3

performance, both in terms of the F_2 measure and the number of *Exact* matches. Furthermore we see that the performance of the implemented methods is comparable to that of the human annotators, reaching average human annotator performance in terms of recall.

Further improvements of the system may include methods for filtering out the meaningless keyphrase candidates, as well as new keyphrase candidate generation methods based on statistical data rather than a predefined set of filters.

Acknowledgments

This work has been supported by the Ministry of Science, Education and Sports, Republic of Croatia under the grant No. 036-1300646-1986. The authors are grateful to the Croatian News Agency (HINA) for making available the newspaper corpus. The authors thank the anonymous reviewer for his or her useful comments.

References

Ahel R.; Dalbelo Bašić B.; Šnajder J. 2009. Automatic keyphrase extraction from Croatian newspaper articles. In *The Future of Information Sciences, Digital Resources and Knowledge Sharing*, pages 207–218.

Bekavac B. & Tadić M. 2007. Implementation of Croatian NERC system. In *Proceedings of the Workshop on Balto-Slavonic Natural Language Processing 2007, Special Theme: Information Extraction and Enabling Technologies*, pages 11–18. Association for Computational Linguistics (ACL).

Medelyan O. & Witten I.H. 2006. Thesaurus based automatic keyphrase indexing. In *Proceedings of the 6th ACM/IEEE-CS Joint Conference on Digital Libraries, 2006. JCDL'06*, pages 296–297.

Turney P.D. 2002. Learning to extract keyphrases from text. *Arxiv preprint cs/0212013*.

Šnajder J.; Dalbelo Bašić B.; Tadić M. 2008. Automatic acquisition of inflectional lexica for morphological normalisation. *Information Processing and Management*, 44(5):1720–1731.

Witten I.H.; Paynter G.W.; Frank E.; Gutwin C.; Nevill-Manning C.G. 1999. KEA: Practical automatic keyphrase extraction. In *Proceedings of the fourth ACM conference on Digital libraries*, page 255. ACM.

Zesch T. & Gurevych I. 2009. Approximate matching for evaluating keyphrase extraction. In *Proceedings of the 7th International Conference on Recent Advances in Natural Language Processing (electronic proceedings)*, pages 484–489, Borovets, Bulgaria.

CORRECTING WORD MERGE ERRORS IN CROATIAN TEXTS

Mladen Mikša, Jan Šnajder and Bojana Dalbelo Bašić

University of Zagreb, Faculty of Electrical Engineering and Computing
Unska 3, Zagreb, Croatia
{mladen.miksa, jan.snajder, bojana.dalbelo}@fer.hr

ABSTRACT

In many text processing tasks character-level errors (due to mistyping, OCR, etc.) typically lead to performance degradation. Most approaches to error correction are dictionary based and cannot be used to correct word boundary errors. Word boundary errors are quite common in OCR-generated texts, especially the word merge errors. In this paper we describe an approach to correcting word merge errors in texts written in Croatian language. The approach is based on combinatorial optimization with beam search strategy that determines the most plausible segmentation of the input token. The plausibility of the segmentation is assessed using a statistical language model and several heuristics. We evaluate the performance of our approach on a sample of artificially generated word merge errors. The achieved results are comparable to the results of the approaches found in the literature.

1. Introduction

With the ever rising amount of gathered information that needs to be analyzed and used for a multitude of diverse purposes, the importance of computers substantially increases. In order for the computers to be able to process the information, it needs to be converted into a digital form. Data encoded in a natural language presents one type of information, which has a wide use in diverse applications. Texts in digital form can be obtained by manual input or by some other automatic means (e.g., optical character recognition, OCR).

Documents containing character errors – either caused by mistyping or introduced by the document acquisition process – present an inconvenience to the reader and a great difficulty for the automatic processing of text. In many text processing tasks (e.g., information retrieval, information extraction, question answering) character-level errors typically lead to performance degradation. In particular, errors introduced by the OCR process often cause problems in further document processing (Cristea et al. 2008; Vojnovski et al. 2005; Wick et al. 2007).

Various approaches for tackling such errors were developed, both for OCR-generated texts (Taghva and Stofsky 2001; Mihov et al. 2003) and texts acquired by other means (Ingels 1996). Most approaches for error correction are based on some form of dictionary lookup and were shown to achieve good performance. However, dictionary-based approaches presuppose that a (possibly incorrect) word token corresponds to a character sequence separated by white spaces. For this reason such approaches cannot be used to correct word boundary errors (i.e., deletion or insertion of a white space), which are quite common in OCR-generated texts. Developing an efficient method for handling word boundary errors turns out to be a difficult task (Kukich 1992; Verberne 2002).

Problems in discerning word boundaries can vary depending on the language in question. Detecting word boundaries is extremely important in Japanese texts, because of the absence of word boundary characters. In highly inflectional languages, such as most Slavic languages, detecting word boundaries is more difficult because of the variety of suffixes that occur in the text. However, some of the language specific characteristics can be used to improve the performance of word segmentation.

In this paper we describe an approach to correcting word boundary errors in texts written in Croatian language, which was initially developed as a part of an OCR error correction system (Marović et al. 2010). We focus on one type of word boundary errors, namely word merge errors, which we found to be prevalent in OCR-generated texts. The analysis of a sample consisting of 81 OCR-generated texts, acquired from the

Croatian News Agency (HINA), showed that word boundary errors accounted for 30.89% of the total number of OCR errors, with word merge errors amounting to 88.13% of boundary errors. Our approach is based on combinatorial optimization with beam search strategy that determines the most plausible segmentation of the input token. The plausibility of the segmentation is assessed using a statistical language model and several heuristics. We present the evaluation of the performance of our approach and argue that its efficiency is comparable with the efficiency of other approaches.

The rest of the paper is structured as follows. In the next section we give a brief overview of some of the methods used in correcting word boundary errors. Section 3 gives a full description of our approach to segmenting merged words and its implementation, while Section 4 presents the results of the evaluation. Section 5 concludes the paper and presents some outlines for future work.

2. Related work

Word boundary errors present a challenging problem for the correction methods. The number of segmentations of the input token is exponential in the token length, making it impossible to check all segmentations in acceptable time. Because checking all possible segmentations is unfeasible, several approaches for finding the most likely segmentation were developed to circumvent the problem. However, many proposed correction systems do not correct word boundary errors and leave their correction to the user. This section presents some of the approaches used in correcting word boundary errors with stronger emphasis on the correction of word merge errors.

Kolak et al. (2003) developed an approach to correcting OCR errors based on inverting a generative probabilistic OCR model. In order to resolve word merge errors, their system tries to segment the input token into two parts at every position in the token. The probability of a segmentation is estimated using a language model. The problem with this approach is in its inability to correct word merge errors consisting of more than two words. The evaluation of the whole OCR correction system was performed and it has showed a decrease in word error rate from 18.31% to 6.75%.

Taghva and Stofsky (2001) built a semi-automatic system for correcting OCR errors. Their approach to word boundary errors consisted in a heuristic word boundary procedure, which corrects the token only if the heuristics are activated. Detailed account of the involved heuristics is not given in the paper, nor is the system performance on word merge errors demonstrated. For the performance measure they used word accuracy, the percent of accurate words in the corrected text. As the system is semi-automatic, the evaluation was performed with user interaction. The system has achieved the improvement in word accuracy from 98.18% to 99.79% for one document and an improvement from 98.46% to 99.85% for the other document. As in (Kolak et al. 2003), separate evaluation of word boundary error correction was not performed.

Nagata (1994) presented a forward-DP backward-A* algorithm for the segmentation of sentences written in Japanese language. The algorithm makes two passes through the input token. In the first pass the algorithm generates all the words that can start from each position in the input with each of the words getting a partial path score. The score is calculated from the best partial path score leading up to the generated word and the probabilities of continuing the path with the generated word. Those probabilities were estimated using the training corpus. The second pass of the algorithm uses those scores as heuristics in the A* search and produces the most likely segmentation. Training and evaluation was performed on a portion of the ATR Dialogue Database (Ehara et al. 1990) using precision and recall as the performance measures. The algorithm achieved precision of 97.2% and recall of 97.7%. This algorithm, with slight modifications, was also used for correcting OCR errors (Nagata 1996).

Ingels (1996) created a system for correcting closed-domain input queries written in Swedish. He used layered Hidden Markov Models (HMM) in which he represented each word type with a single word model (584 models in total). Models represent correct words of the Swedish language along with errors that typically

occur in them. The correct and erroneous character sequences in the models are given probabilities of occurrence, which were trained on a corpus with artificially generated errors. All of the word models were then interconnected to provide contextual information that is used to limit the search space and improve the results. Because of such interconnections between models, the system can correct word merge errors. The correction works by scanning the input from left to right and making appropriate transitions through the models. Beam search is used in order to keep the number of candidates manageable. Evaluation was performed on a data set of textual queries in terms of precision and recall. For word merge errors the system achieved precision of 87% and recall of 100%. As noted by Ingels, while the approach demonstrated good results, it is unlikely to be practical for unrestricted (open-domain) text.

The approach described in this paper is based on combinatorial optimization, enabling it to correct merges of multiple words, unlike the approach described in (Kolak et al. 2003). However, because of the beam search strategy, we do not have the guarantee that the result is optimal, which presents a downside when compared to (Nagata 1994). On the other hand, we suspect that our algorithm runs faster and, judging by the evaluation results, the lack of guarantee seems not to present a practical problem. Ingels (1996) used layered HMMs in order to correct both segmentation and character errors, while our approach tackles only the segmentation problem. On a conceptual level there is a slight similarity between his and our approach in that both approaches are based on the left to right scan of the input token, using beam search in order to limit the number of candidates. However, the details pertaining to the exact use of the beam search, definition of what constitutes a search candidate, and the means by which the score of a candidate is calculated strongly differ. The comparison between our approach and the one used in (Taghva and Stofsky 2001) is not possible because of the incompleteness in the description of their approach.

3. Segmenting merged words

Our approach to correcting word merge errors (i.e., segmenting merged words) is based on a segmentation algorithm. Because checking all possible segmentations is not acceptable, the algorithm uses the beam search strategy, i.e., it considers only a constant number of segmentation candidates at each step. In this section we give a detailed description of the algorithm and its implementation, along with the description of the language model that was used as the basis for determining the scores of segmentation candidates.

3.1. The segmentation algorithm

The algorithm is based on a beam search strategy that searches through possible segmentations in order to optimize the segmentation candidates score. The basis for these scores is provided by the language model (described in the subsection 3.3.), while several Croatian language specific heuristics are used for the additional improvement of the results. Pseudocode in Fig. 1 presents the core of the algorithm, while an example of its execution is given in Table 1. The algorithm works by processing the input token from left to right, one character at a time while keeping N-best scored segmentation candidates found up to that point.

The inputs of the algorithm are the token and its left context. The token (*token*) represents a unit of the processed text and may be a valid or erroneous word (a string delimited by white spaces), `HTML` tag, or punctuation mark (among others). Only the word tokens that are not contained in the dictionary are processed by the algorithm. This is a limitation because the algorithm cannot correct merge errors that result in a valid word. Also, if some correct word is not contained in the dictionary, the algorithm will try to segment it, which may result in the corruption of the original word. Additionally, a condition on the token length can be set, so that the segmentation algorithm process only those tokens whose length equals or exceeds a predetermined threshold. Left context (*leftContext*) contains the tokens directly preceding the one contained in the input variable *token*. It is used in scoring the segmentation candidate via the language model. Segmentation candidates are stored in the local array *candidates* of fixed size. The algorithm returns the best segmentation candidate and its score divided by the number of newly segmented words. The correction is

Input: *token* – the token that is being segmented.
leftContext – left context of the token.
Output: *candidate* – segmentation result and its score.
Local: *candidates* – an array of segmentation candidates of fixed size *N*.

candidates := generateInitial(*token*[1], *leftContext*)

for *i* := 2 **to** length(*token*) **do**
segment(*candidates*, *token*[*i*])
progress(*candidates*, *token*[*i*])
end for

return chooseCandidate(*candidates*)

Figure 1: Segmentation algorithm

Table 1: Execution of the segmentation algorithm (example)

Iteration (*i*)	Executed command	Array sorted by candidate score (N = 3)
0	*token* = "jedosta" *leftContext* = ". Bilo"	$\emptyset$
1	generateInitial	[(j, 0)]
2a	segment	[(j, 0), (j|, −4.81)]
2b	progress	[(je, 0), (j|e, −4.81)]
3a	segment	[(je|, 0.85), (je, 0), (j|e, −4.81)]
3b	progress	[(je|d, 0.85), (jed, 0), (j|ed, −4.81)]
4a	segment	[(je|d, 0.85), (jed, 0), (je|d|, −2.28)]
4b	segment	[(je|do, 0.85), (jedo, 0), (je|d|o, −2.28)]
5a	segment	[(je|do, 0.85), (jedo, 0), (je|do|, −1.66)]
5b	progress	[(je|dos, 0.85), (jedos, 0), (je|do|s, −1.66)]
6a	segment	[(je|dos, 0.85), (jedos, 0), (je|do|s, −1.66)]
6b	progress	[(je|dost, 0.85), (jedost, 0), (je|do|st, −1.66)]
7a	segment	[(je|dost, 0.85), (jedost, 0), (je|do|st, −1.66)]
7b	progress	[(je|dosta, 0.85), (jedosta, 0), (je|do|sta, −1.66)]
return	chooseCandidate	[(je|dosta|, 0.29)]

accepted if the candidate score exceeds a predetermined threshold.

The algorithm works as follows. It starts by adding the initial candidate to the *candidates* array (generateInitial). The initial candidate contains the left-most character of the input token (*token*[1]) and its left context. Main part of the processing is done in the for loop that iterates through the characters of the token starting with the second character. Each iteration is divided into two steps: segment and progress. At the segment step, every candidate contained in the array is segmented at the current point creating a new candidate, which is then scored and added to the array. If the limit *N* in the number of candidates has been reached, the size of the array is maintained by eliminating the lowest scored candidate. After the segmentation, the progress step adds the current character (*token*[*i*]) to all the remaining candidates.

The example in Table 1 shows the execution of the algorithm for the token "jedosta", consisting of two valid words: "je" (is) and "dosta" (enough), with the value of left context set to ". Bilo" (was). The execution is carried out through seven iterations, pertaining to the number of characters of the input token, and with the array limited to three elements. For each iteration the executed command and the resulting array are given. Elements of the array are presented as a pair consisting of a partially segmented token (where the character '|' represents the word delimiter) and its score. An example of discarding a segmentation candidate can be

seen at the step 3a where the segmentation candidate "j|e|", the result of segmenting "j|e", gets discarded.

3.2. Candidate scoring

The score of the new segmentation candidate is calculated by taking the score of the original candidate and adding to it the score of a newly segmented word. The score of the new word is the sum of the logarithm of language model probability p_{LM} and the defined heuristic values. Three heuristics are used in the algorithm and are based on valid words, suffixes, and character changes. In case that the newly segmented word is a valid word, heuristic value h_{word} is added to its score. A list of suffixes is used to additionally differ among the candidates. Normalized chi-squared (χ^2) value is given to each suffix as a measure of its reliability: a higher χ^2 value indicates that the string is a reliable suffix, whereas a lower χ^2 value indicates that the string can also appear as a non-suffix. The new word is searched for the suffix with the highest reliability and its value, weighted with the heuristic value h_{suffix}, is added to the words score. Heuristics h_{char} uses the idea that the letter case does not change in the middle of a word and that the words consist only of letters. Thus, if the character following a newly segmented word is not a letter or it changes case, the heuristic value h_{char} is added to the words score. The final equation for the score of a new candidate $score_{n+1}$ is thus given by:

$$score_{n+1} = score_n + \log p_{LM} + h_{word} + h_{suffix} \cdot \chi^2 + h_{char}.$$

3.3. Statistical language model

A statistical language model is used for the estimation of candidate probabilities. Language models are inspired by the idea that it is possible to assign probabilities to sentences. Let $w_1^n = w_1 \ldots w_n$ denote the sequence of words of a sentence. Then, using the chain rule of probability we can formulate the probability of a sentence w_1^n as (Jurafsky et al. 2000):

$$P(w_1^n) = P(w_1)P(w_2|w_1)P(w_3|w_1^2)\cdots P(w_n|w_1^{n-1}) = \prod_{k=1}^{n} P(w_k|w_1^{k-1}).$$

Estimating and storing all such probabilities is impossible in practice, so we use the Markov assumption. It states that for calculating the probability of a current word we only need to look at the finite number N of words preceding it. We call such a model the Nth-order Markov model or $(N+1)$-gram, where the approximation is given by:

$$P(w_n|w_1^{n-1}) \approx P(w_n|w_{n-N}^{n-1}).$$

In the case of the bigram model (first-order Markov model) the probability of a full sentence is given by:

$$P(w_1^n) \approx \prod_{k=1}^{n} P(w_k|w_{k-1}),$$

where $w_0 = \langle s \rangle$ represents the start of a sentence. In our approach we use a trigram model.

We use modified Kneser-Ney smoothing for the better estimation of the probabilities of N-grams with low or zero frequency in the training corpus. For details of Knesery-Ney smoothing, the reader is referred to (Chen and Goodman 1998). Using word classes instead of distinct word forms can make the model more general and robust. Because of that, we represented numbers in the corpus with their own class denoted by the tag *<num>* and all punctuation marks with a class denoted by *<pun>*. The language model was built with the *SRI Language Modeling Toolkit*[1] (Stolcke 2002).

[1] `http://www-speech.sri.com/projects/srilm/`

4. Evaluation

This section presents the performance evaluation of the described approach. The algorithm and the corresponding components where implemented in the C# programming language. Acquiring large data sets consisting of erroneous texts generated by real world applications, and their corresponding corrected versions, presents a time consuming task. Also, such data sets would contain other types of character errors, whose correction is not covered by the approach presented in this paper. In order to counter these problems, the evaluation was performed on a data set with artificially generated word merge errors, which simulate errors found in real applications.

4.1. Training and test data

As training and test data we used two newspaper corpora: *Vjesnik* and *Glas slavonije*. For building and evaluating the language model we used the articles published in the daily Croatian newspaper *Vjesnik* (totaling around $4 \cdot 10^6$ sentences and 10^8 words). The corpus was split into two uneven subsets; the first part ($9/10$ of the corpus) was used for building the language model, while the second part was used for model evaluation. The unigram part of the language model was additionally expanded with words acquired from an inflectional lexicon (those words were assigned the probability of an unknown word as calculated by the SRILM). The lexicon was acquired from the articles of *Vjesnik* and the Official Gazette of the Republic of Croatia using the procedure described in (Šnajder et al. 2008). Since the procedure is automatic, it introduces a number of morphologically invalid word forms into the model. However, as merges between words are unlikely to produce such an invalid word form, lexicon errors should not present a problem for word merge error correction. The suffixes were extracted from the aforementioned lexicon.

For the performance evaluation of the approach we used a sample consisting of randomly chosen sentences from the newspaper *Glas Slavonije*. As the sampled sentences are mostly error-free, similar to (Nagata 1996; Wick et al. 2007) we artificially generated word merge errors as follows. Around 5% of the white space characters were deleted from the sentences. Additional merges of two to a maximum of seven words (number of merges is chosen randomly) were performed every thousand characters on average. This merges were performed in order to simulate the merge of multiple words that we found to be common in OCR texts. The sample was divided into two disjunct subsets: the first subset (totaling 139,438 tokens) was used for parameter optimization, whereas the second subset (totaling 154,866 tokens) was used for the evaluation.

4.2. Language model evaluation

Commonly used measures for the evaluation of language models are the *cross entropy* and *perplexity*. The cross entropy determines the quality of the model distribution m as an approximation of the actual probability distribution p. Lower values of cross entropy yield a better model, although cross entropy can not be lower than the entropy of the actual distribution p. The equation for cross entropy is given by (Jurafsky et al. 2000):

$$H(p, m) = \lim_{n \to \infty} \frac{1}{n} \sum_{W \in L} p(w_1, \ldots, w_n) \log m(w_1, \ldots, w_n).$$

Perplexity is given by 2^H and can intuitively be viewed as the average number of words between which we can choose the next word of a sentence (Jurafsky et al. 2000).

Evaluation of the language model yielded the perplexity value of 340.09. This result is worse than those found in the literature; e.g., Chen and Goodman (1998) reported the perplexity value of around 250 for different kinds of language models for English language. This demonstrates that there is room for improvement of the model, although the obtained result can also be a consequence of the higher morphological complexity of the Croatian language.

4.3. Performance measures

Many measures for evaluating error correction have been proposed in the literature. Following (Ingels 1996; Nagata 1994) in this work we use standard information retrieval measures of precision, recall, and F_1 score. We counted the number of words in the corrected texts, N_{cor}, number of words in the accurate (error-free) texts, N_{acc}, and the number of matched words between these two sets, N_{mat}. Precision defines the percent of the corrected texts that match the accurate counterparts and is given by the equation $P = N_{mat}/N_{cor}$. Recall defines the completeness of the correction, that is the percent of the accurate texts contained in the corrected texts. The equation for recall is given by $R = N_{mat}/N_{acc}$. The F_1 score combines recall and precision using the harmonic mean and gives the overall score of the test, $F_1 = 2PR/(P + R)$.

4.4. Results

Table 2 shows the results of the parameter tuning performed on a training set (only results for some selected parameter values are shown). The tuning was performed by an extensive manual search. Rows pertaining to the change of one parameter or group of parameters are separated from each other so as to make it easier to note their influence; parameters that are being changed are typeset in bold. We performed extensive tests in order to find the parameters that optimize the F_1 score. Table 3 shows the evaluation results on previously unseen data using the optimal parameters found in the training stage.

Parameter tuning was performed on a selection of sentences containing artificially generated word merge errors. The sentences had the initial values (i.e., initial text quality) of precision of 95.19%, recall of 90.60%, and F_1 score of 92.84%. The results show a substantial increase in both precision and recall (cca. +2.5% and +7%, respectively). Use of heuristics showed a slight deterioration in precision, which was compensated by the significant increase in recall. Valid word heuristics (h_{word}) had the greatest impact on the results tending to bias the algorithm towards performing more segmentations. This resulted in recovering a greater amount of the original text, although it introduced more errors in the process. The impact of suffix (h_{suffix}) and character level (h_{char}) heuristics is less significant. Suffix heuristics showed the greatest preservation of precision, while still achieving an increase in recall. Character heuristics had the least significant influence, which is expected because of the special cases that they handle.

Setting the token length threshold to higher values deteriorated the results, indicating that the segmentation performs well even when two shorter words have been merged. However, as shown in Table 2, setting the length threshold to too low values also had a negative effect on the results, because of the errors it introduced. Finally, tests pertaining to the size of the candidate array (N) show that the size of array does not influence the performance, thus indicating that the best segmentation candidates are consistently top-scored.

Evaluation on unseen data was performed with the optimal parameters found in the tuning stage (Table 3). The algorithm demonstrated an improvement in precision from 95.07% to 97.62% (+2.55%), in recall from 90.40% to 97.48% (+7.08%), and in F_1 score from 92.68% to 97.55% (+4.87%). This amounts to correctly segmenting additional 10,966 words of the initial data, while 3,897 wrongly merged words were either incorrectly segmented or left unsegmented.

Comparing our results to those found in literature is somewhat problematic because of the differences in languages and the measures used for the evaluation. Taghva and Stofsky (2001) and Kolak et al. (2003) developed approaches to correcting OCR errors, but did not perform separate evaluation of the word merge errors correction, making the comparison impossible. Nagata (1994) and Ingels (1996) use precision and recall for the evaluation of word merge error correction. Although the definitions of these measures somewhat differ between them, as well as between the definition used in this work, the differences are small enough to allow for a comparison. Nagata (1994) reports the precision value of 97.2% and recall of 97.7%, which are comparable to the results of our approach (note that Nagata's approach segments whole sentences rather than their incorrectly merged parts). Ingels (1996) reports the precision of 87% and recall of 100%, which

Table 2: Parameter tuning on training data (initial text quality: P = 95.19%, R = 90.60%, F_1 = 92.84%)

Threshold			Heuristics					
length	score	N	h_{word}	h_{suffix}	h_{char}	P (%)	R (%)	F_1 (%)
1	−20	50	**0.0**	**0.0**	**0.0**	97.73	96.59	97.16
1	−20	50	**1.1**	0.0	0.0	97.57	97.45	97.51
1	−20	50	0.0	**3.5**	0.0	97.73	96.85	97.29
1	−20	50	0.0	0.0	**0.9**	97.65	96.78	97.22
1	−20	50	**1.1**	**0.5**	**0.6**	97.51	97.53	97.52
1	−20	50	1.1	0.5	0.6	97.51	97.53	97.52
4	−20	50	1.1	0.5	0.6	97.61	97.58	97.59
10	−20	50	1.1	0.5	0.6	97.08	95.26	96.16
4	**−20**	50	1.1	0.5	0.6	97.61	97.58	97.59
4	**−5**	50	1.1	0.5	0.6	97.61	97.58	97.59
4	−6	**40**	1.1	0.5	0.6	97.61	97.57	97.59
4	−6	**1000**	1.1	0.5	0.6	97.61	97.57	97.59

Table 3: Evaluation on the test data (initial text quality: P = 95.07%, R = 90.40%, F_1 = 92.68%)

Threshold			Heuristics					
length	score	N	h_{word}	h_{suffix}	h_{char}	P (%)	R (%)	F_1 (%)
4	−6	50	1.1	0.1	0.7	97.62	97.48	97.55

demonstrates better recall than our approach, but significantly lower precision. Not much work has been done on the topic of word segmentation and what was done was mostly incorporated as a part of a larger system. Thus, overall, our approach seems to be of comparable performance to other published approaches.

5. Conclusion

Correcting errors in documents is an important step in many automatic text processing tasks. Word boundary errors present an especially challenging problem for text correction. In this paper, we presented an approach to correcting word merge errors (a type of word boundary errors) in texts written in Croatian language, based on a combinatorial optimization with beam search strategy. Evaluation performed on a test data with artificially generated word merge errors showed the increase in F_1 score from 92.68% to 97.55%. The achieved results are comparable to the results of the approaches found in the literature. With appropriate changes in the language model and heuristics, our approach can also be applied to other languages.

In the future, we plan on experimenting with different types of language models, which may lead to an increase in the performance. Also, as the use of heuristics showed some promise, devising new heuristics could further improve the performance of the approach. A special problem of our approach pertains to the correct words that are not contained in the dictionary. Building some form of a local dictionary, i.e., a dictionary built from the document that is being corrected, could alleviate the problem of unknown words. Although our approach yields satisfactory results when compared to those found in the literature, we are considering implementing the forward-DP backward-A* algorithm of Nagata (1994).

Acknowledgments

This work has been supported by the Ministry of Science, Education and Sports, Republic of Croatia under the Grant 036-1300646-1986. The authors thank the anonymous reviewer for his or her useful comments.

References

Chen S. & Goodman J. 1998. An empirical study of smoothing techniques for language modeling. *Computer Speech and Language*, 13(4):359–393.

Cristea D.; Forăscu C.; Răschip M.; Zock M. 2008. How to Evaluate and Raise the Quality in a Collaborative Lexicographic Approach. *LREC-2008, Marakesh*.

Ehara T.; Ogura K.; Morimoto T. 1990. ATR dialogue database. In *First International Conference on Spoken Language Processing*. ISCA.

Ingels P. 1996. Connected text recognition using layered HMMs and token passing. In *Proceedings of the Second Conference on New Methods in Language Processing*, pages 121–132.

Jurafsky D.; Martin J.H.; Kehler A.; Vander Linden K.; Ward N. 2000. *Speech and language processing*. Prentice Hall New York.

Kolak O.; Byrne W.; Resnik P. 2003. A generative probabilistic OCR model for NLP applications. In *Proceedings of the HLT-NAACL*, volume 3.

Kukich K. 1992. Techniques for automatically correcting words in text. *ACM Computing Surveys (CSUR)*, 24 (4):439.

Marović M.; Mikša M.; Šnajder J.; Dalbelo Bašić B. 2010. Croatian OCR Error Correction Using Character Confusions and Language Modelling. In *Proc of the Central European Conference on Information and Intelligent Systems, CECIIS 2010*, *(in press)*.

Mihov S.; Koeva S.; Ringlstetter C.; Schulz K.U.; Strohmaier C. 2003. Precise and efficient text correction using Levenshtein automata, dynamic Web dictionaries and optimized correction models. In *Proc of the 1st Int Workshop on Proofing Tools and Language Technologies. Patras, Greece: Patras University*.

Nagata M. 1994. A stochastic Japanese morphological analyzer using a forward-DP backward-A* N-best search algorithm. In *Proceedings of COLING*, volume 94, pages 201–207.

Nagata M. 1996. Context-based spelling correction for Japanese OCR. In *Proc. of the 16th COLING*, pages 806–811.

Stolcke A. 2002. SRILM – an extensible language modeling toolkit. In *Seventh International Conference on Spoken Language Processing*, volume 3, pages 901–904.

Šnajder J.; Dalbelo Bašić B.; Tadić M. 2008. Automatic acquisition of inflectional lexica for morphological normalisation. *Information Processing and Management*, 44(5):1720–1731.

Taghva K. & Stofsky E. 2001. OCRSpell: an interactive spelling correction system for OCR errors in text. *International Journal on Document Analysis and Recognition*, 3(3):125–137.

Verberne S. 2002. Context-sensitive spell checking based on word trigram probabilities. *Master's thesis, University of Nijmegen*.

Vojnovski V.; Džeroski S.; Erjavec T. 2005. Learning PoS tagging from a tagged Macedonian text corpus. *Proceedings of SIKDD 2005*.

Wick M.L.; Ross M.G.; Learned-Miller E.G. 2007. Context-sensitive error correction: using topic models to improve OCR. In *Ninth International Conference on Document Analysis and Recognition, 2007. ICDAR 2007*, volume 2.

ASSESSING THE FEATURE-DRIVEN NATURE OF SIMILARITY-BASED SORTING OF VERBS

Pinar Öztürk, Mila Vulchanova, Christian Tumyr, Liliana Martinez, David Kabath

Department of Linguistics, Faculty of Philosophy, University of Zagreb
Ivana Lučića 3, 10000 Zagreb, Croatia
marko.tadic@ffzg.hr

ABSTRACT

The paper presents a computational analysis of the results from a sorting task with motion verbs in Norwegian. The sorting behavior of humans rests on the features they use when they compare two or more words. We investigate what these features are and how differential each feature may be in sorting. The key rationale for our method of analysis is the assumption that a sorting task rests on a similarity assessment process. The main idea is that a set of features underlies this similarity judgment, and similarity between two verbs amounts to the sum of the weighted similarity between the given set of features. The computational methodology used to investigate the features is as follows. Based on the frequency of co-occurrence of verbs in the human generated cluster, weights of a given set of features are computed using linear regression. The weights are used, in turn, to compute a similarity matrix between the verbs. This matrix is used as an input for the multidimension scale clustering and agglomerative hierarchical clustering. If the selected/projected set of features aligns with the features the participants used when sorting verbs in groups, then the clusters we obtain using this computational method would align with the clusters generated by humans. Otherwise, the method proceeds with modifying the feature set and repeating the process. Features promoting clusters that align with human-generated clusters are evaluated by a set of human experts and the results show that the method manages to identify the appropriate feature sets. This method can be applied in analyzing a variety of data ranging from experimental free production data, to linguistic data from controlled experiments in the assessment of semantic relations and hierarchies within languages and across languages.

Key words: verb clustering, features, weighting, hierarchical clustering

1. Introduction

Sorting tasks are a popular knowledge elicitation technique used in psychology and cognitive studies [3], [5]. In a typical sorting task participants are asked to sort in groups items in a particular domain. This kind of task rests on the common assumption that, in categorization processes, humans rely on specific features that differentiate one group of objects from another, and that these features characterize and define the group in a broader domain ([6]).

We designed a sorting task to study the semantic domain of verbs of human locomotion below the basic level ([8], [9], [4]). Specific verbs of locomotion include words, such as English strut, stroll, gambol, hop, and the like.

Our main assumption is that the way speakers group those verbs is revealing about the semantic structure of this field. Our hypothesis is that the size (how many) and constitution (what verbs) of these groups can be used to derive the semantic features that characterize both individual lexical items and the domain as a whole. We investigated whether and how it is possible to discover such relations and patterns for the set of motion related verbs, based on verb clusters provided by the human subjects. The paper presents a computational method that aims to discover the most salient features and their degree of saliency.

The outline of the paper is the following: We first introduce the human sorting task experiment and its linguistic background. We then proceed with the computational method, the computational experiment and the results of applying this method. We conclude with a discussion of the results obtained and provide a summary.

2. Human experiments

Germanic languages are characterized by a rich system of specific verbs describing locomotion, and the distinctions among the items in this domain are not always very clear. Furthermore, little is known about the way native speakers of these languages acquire such highly specific vocabulary, and whether they use salient perceptual features of the actions these words denote, and then map these features onto the lexical items at hand or simply rely on the linguistic contexts in which they first encounter these verbs [7].

As a first step in studying the native speakers knowledge of specific locomotion verbs, we asked native speakers of Norwegian to group 41 verbs that were selected through a 3 step process, a semantic recall task, an elicitation task, with results from both being compared to a comprehensive list compiled on the basis of dictionary information [4].

The verbs appeared on small paper cards and participants were asked to sort them in groups by similarity. Participants are then asked to describe what features they have used in the grouping process. All the features mentioned by one or several subjects constitute the candidate feature set having 15 features. Using the computational method described in the next section, we tried to select the subset from this candidate set of the features that were most influential in the overall sorting experiment.

To avoid confounding of the results, we asked participants to remove all words whose meaning they did not know. The groups for each participant were photographed by digital camera, and the results for all participants were manually entered in an excel file and consequently converted into a verb co-occurrence matrix of which each cell indicates how many of the subjects put the two corresponding verbs into the same group. These raw data served as the input for two kinds of analyses, multi-dimensional scaling and agglomerative hierarchical clustering. This matrix constitutes also the input to the computational method described in the next section.

3. The computational method

The inputs to the method are the candidate feature set gathered from the human subjects as described in the preceding section, and the verb co-occurrence matrix prepared after the human experiment.

When asked to group the set of verbs, the human subjects had the opportunity of placing verbs whose meaning they did not know or for some reason whose placement they felt uncertain about in a separate group labeled "out", which indicated exclusion from the sorting. Verbs excluded in this way are considered as a negative contribution and were excluded from further analyses. A total of three verbs were excluded by more than two subjects and were removed from the dataset for analysis.

The overall method is summarized in Algorithm 1. The algorithm describes the process of evaluating the calculated feature weights with regard to the data provided by the human subjects. The grouping data provided by the human subjects are clustered (the result is denoted as C_{human} in Algorithm 1 using agglomerative hierarchical clustering. After the weights of the features are computed as explained in section 3.1, a verb similarity matrix is computed (explained in section 3.2) using these weights. Then, using this new weight-based similarity matrix the verbs are clustered using the same clustering methods. These clusters are depicted as C_{comp} in Algorithm 1). S_{human} is a verb-verb matrix resulting from the human sorting of the verbs, and considered to represent the human judgment of similarity between the verbs.

If the computed clusters C_{comp} and human based clusters C_{human} align, i.e. are fairly similar (depicted as $C_{comp} \cong C_{human}$), the features and weights are considered to indicate what the human subjects based their clustering of the verbs on. If the clusters do not align, the features with the lowest weight are removed from the set of features, and the process is repeated until an alignment has been achieved.

Algorithm 1 Method

1: $C_{human} \leftarrow$ Cluster data based on S_{human}
2: **repeat**
3: Generate feature-based verb similarity matrix S_f
4: Compute feature weights **W** as described in algorithm 2
5: Generate weighted feature-based verb similarity matrix S_{comp} using $\boldsymbol{W}$ and S_f
6: $C_{comp} \leftarrow$ Cluster the data based on S_{comp}
7: Evaluate alignment between C_{human} and C_{comp}
8: if $C_{comp} \neq C_{human}$ then
9: Remove the feature with the lowest weight
10: end if
11: **until** $C_{comp} \cong C_{human}$ **or** # of features < 2

3.1. Computation of weights

A central idea underlying the proposed method is that similarity between two verbs is equal to the weighted sum of the similarities between the involved features, which is designed by Equation 1.

$$S(v_i, v_j) = w_1 f(a_{1i}, a_{1j}) + w_2 f(a_{2i}, a_{2j}) + \cdots + w_n f(a_{ni}, a_{nj}) \quad (1)$$

where w_n is the weight of feature a_n. The f function uses one of the well-known similarity measures for binary vectors [1]. In addition to the rationale captured by Equation 1, Equation 2 conveys another central assumption in our method:

$$S_{comp}(v_i, v_j) = S_{human}(v_i, v_j) \quad (2)$$

where S_{human} is the verb co-occurrence matrix generated by accumulating the sorting data provided by the subjects. $S_{human}(v_i, v_j)$ represents the number of subjects who put these verbs into the same group. S_{comp} is the computed (more precisely, to be computed) feature-based similarity matrix. A similar approach is taken in [2] where the concerned items are movies and similarity between two movies is associated with the number of persons who rated both of these movies.

The instantiation of equations 1 and 2 for all verbs yields the following linear system of equations, which, when solved, provide values for the weights $w_{1\cdots n}$ for the features.

$$\begin{bmatrix} f(a_{11}, a_{12}) \\ f(a_{11}, a_{13}) \\ f(a_{11}, a_{14}) \\ \vdots \\ f(a_{nm}, a_{nm-1}) \end{bmatrix} \cdot \begin{bmatrix} w_1 \\ w_2 \\ w_3 \\ \vdots \\ w_n \end{bmatrix} = \begin{bmatrix} S_{human}(v_1, v_2) \\ S_{human}(v_1, v_3) \\ S_{human}(v_1, v_4) \\ \vdots \\ S_{human}(v_m, v_{m-1}) \end{bmatrix}$$

Algorithm 2 describes the process of calculating the weights of the 17 features in the candidate feature set. It uses the feature-based verb similarity matrix S_{comp}, and the human generated verb co-occurrence matrix S_{human} to calculate the weights.

Algorithm 2 Calculation of weights

1: $n \leftarrow$ *Number of features*
2: $EQ \leftarrow$ *Empty set of linear equations*
3: **for each** *verb* v_i **do**
4: **for each** *verbs* (v_i); $j = (i + 1)$ **do**
5: *Add* $w_1 f(a_{1i}, a_{1j}) + \cdots + w_n f(a_{ni}, a_{nj}) = S_{human}(v_i, v_j)$ *to* EQ
6: **end for**
7: **end for**
8: *Solve* EQ *for* W
9: **return** $|W|$

The value of feature a_n for a verb v_i is denoted as a_{ni}. The similarity of feature an between verb v_i and v_j is computed by $f(a_{ni}, a_{nj})$, and w_n is the weight or importance of feature a_n. The value of the weights are determined by solving the set EQ of $\frac{i^2}{2}$ linear equations where m denotes the number of verbs. $S_{human}(v_i, v_j)$ denotes the number of subjects having placed the verbs i and j in the same group. The weights W are given as absolute values.

3.2. Computation of feature-based verb similarity matrix

Algorithm 3 describes the process of calculating the similarity between verbs based on the feature weights which were computed using algorithm 2. The similarity between two verbs i and j, denoted as $S_{comp}(v_i, v_j)$ is then computed using Equation 1. This process generates the feature-based similarity matrix S_{comp}.

Algorithm 2 Calculation of verb similarity based on weights

1: $n \leftarrow$ *Number of features*
2: **for each** *verb* v_i **do**
3: **for each** *verb* (v_j); $j = (i + 1)$ **do**
4: $S(v_i, v_j) = \sum_{k=1}^{n} w_k f(a_{ki}, a_{kj})$
5: $S_{comp}(i, j) = S(v_i, v_j)$
6: **end for**
7: **end for**
8: **return** S_{comp}

---

4. Experiments and results

We have conducted a set of experiments to see how the different distance metrics would affect the clustering performance, and the effect of the different linkage methods in hierarchical clustering of the verbs. Another set of experiments were devoted to investigating which features are most salient in the clustering. For this purpose we used the algorithm 2 described in section 3.1 to determine the weights of features and algorithm 3 (in section 3.2) to compute the distance matrix. Then we applied hierarchical clustering, again using different linkage methods.

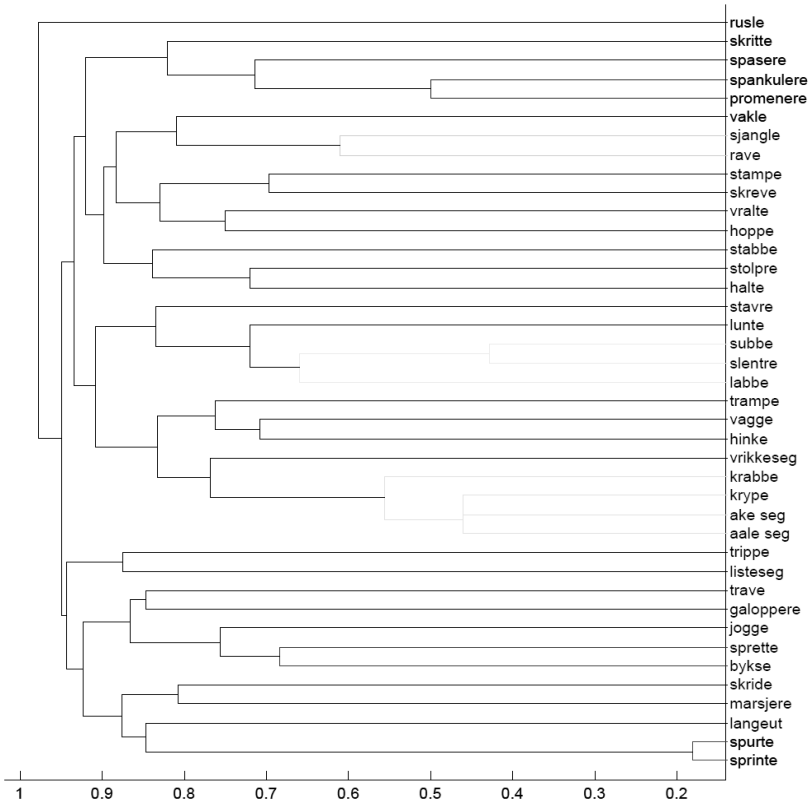


Fig. 1. Clustering of human grouping data using Jaccard metric and Average link.

Regarding the human clustering, we have experimented with the distance metrics provided by MATLAB such as Jaccard, Correlation, Euclidean, Minkowski, Cosine, Chebychev etc. In addition, we have implemented the Multiset distance metric[1] which has proven appropriate in previous analyses of verb similarity [9]. As to linkage methods, MATLAB provides several methods including Centroid, Median, Single, Average, and Complete. The best clustering tree of human grouping data was found to be provided by Euclidean as the distance metric and Average as the linking method. Figure 1 illustrates Jaccard-Average combination while Figure 2 shows the cluster tree when Euclidean-Average combination is used.

We have identified a set of features to have a role, in various degrees, in the human grouping process. Our anticipation is based on the dictionary definitions of the verbs as well as human expert judgments. Using the method presented in section 3 we have estimated the weights (i.e., salience) of the features in the grouping process and then we computed the distance matrix (i.e, the verb-verb matrix) to be used as input for the clustering. We have, experimented with different distance metrics and linking methods.

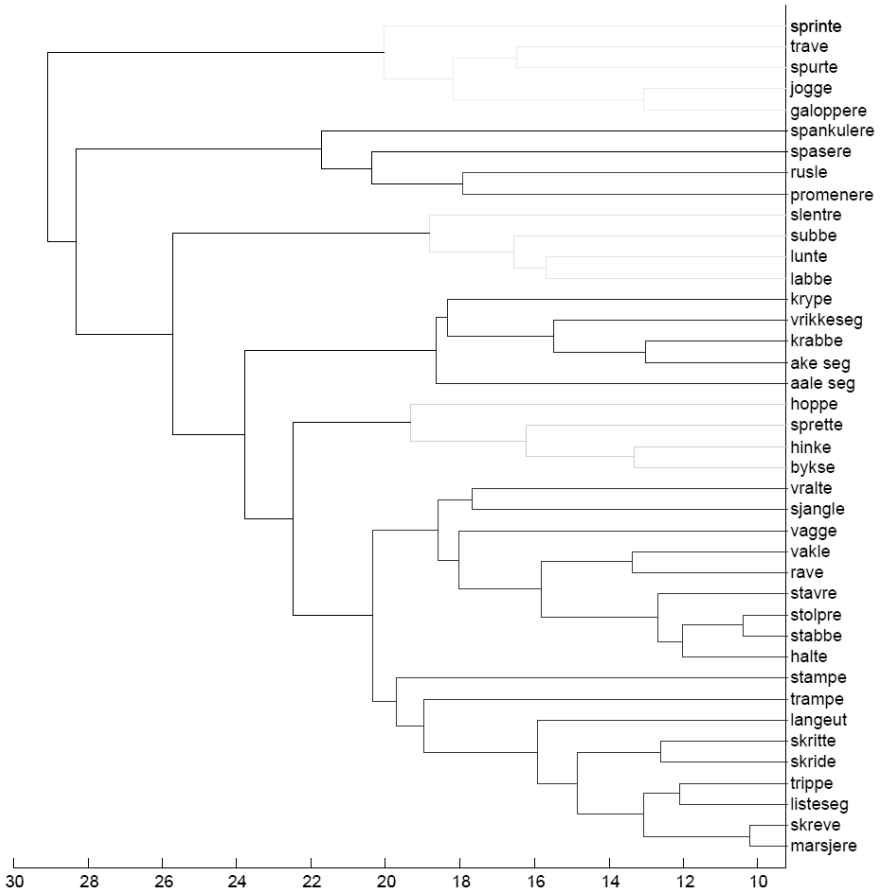


Fig. 2. Clustering of human grouping data using Euclidean metric and Average link.

Initially we had 15 features: *contact* (with substrate), *limbs* (body parts involved in moving), *propulsion* (pattern), *position* (of parts of the body not involved in the motion), *symmetrica*l (motion pattern), *sideways* (motion pattern), *stride* (length), (typical) *agent*, *cause*, *sound* (effects), *speed*, *effort*, *agility*, *social* (context), *purpose*. The computed weights of these features are shown in figure 3. Using these weights we studied the hierarchical trees of the verbs. The Euclidean-Average

[1] $$d(s_i, s_j) = 1 - \frac{\sum_{o \in O} \min(n_o, s_i, n_o, s_j)}{\sum_{o \in O} \max(n_o, s_i, n_o, s_j)}$$

combination has shown the best performance, according to human expert judgments. Two of the hierarchical trees based on these weights are shown in Figures 4 (using Jaccard metric and Average linkage method) and 5 (Euclidean and Average).

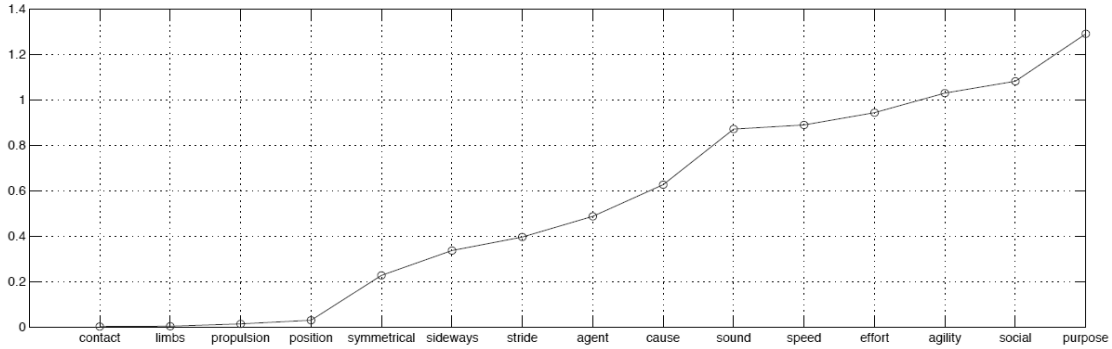


Fig. 3. Weight values of the 15 features

As can be seen in the sorted feature set according to weights (see Figure 3), some of the weight values are significantly lower than the others. Moreover, both the Jaccard Average and the Euclidian Average clusters based on 15 features were not particularly successful in capturing the structure of the semantic field and deviate substantially from the human data cluster, as judged by human experts. Therefore we have analyzed different and fewer numbers of feature combinations. The feature weights showed the same trend while clustering performance varied depending on the number of features and which features were chosen. Figure 6 illustrates the weights for these 9 features: '*contact*', '*limbs*', '*symmetrical*', '*sideways*', '*stride*', '*agent*', '*speed*', '*effort*', '*agility*'.

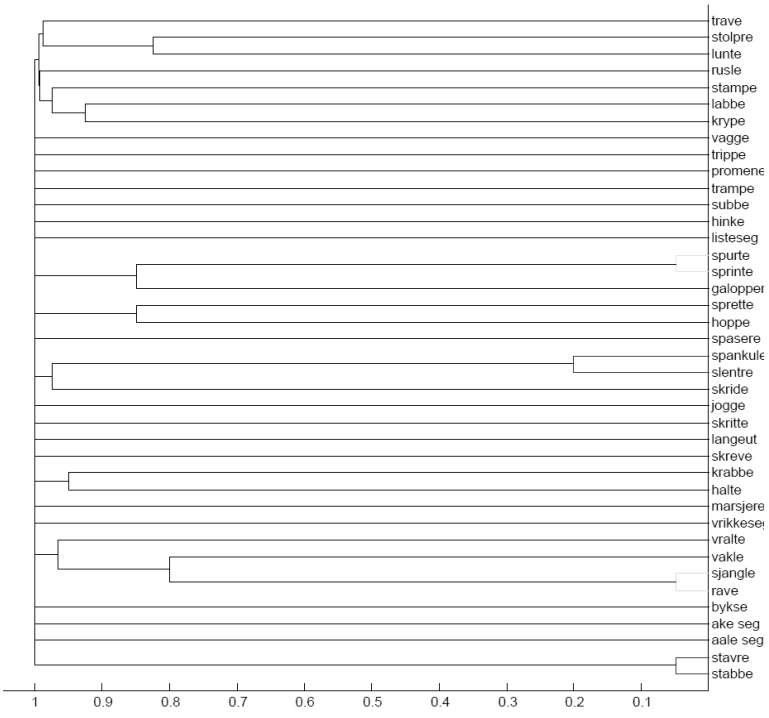


Fig. 4. Clusters based on 15 features, Jaccard metric and Average linkage was used

The clusters based on these 9 features are illustrated in Figures 7 (Jaccard-Average combination), 8 (Correation-Average) and 9 (Euclidean-Average). Figure 10 illustrates the clusters for the following 8 features: '*contact*', '*limbs*', '*symmetrical*', '*sideways*', '*stride*', '*agent*', '*speed*', and '*agility*' where Euclidean metric and Average linking is used.

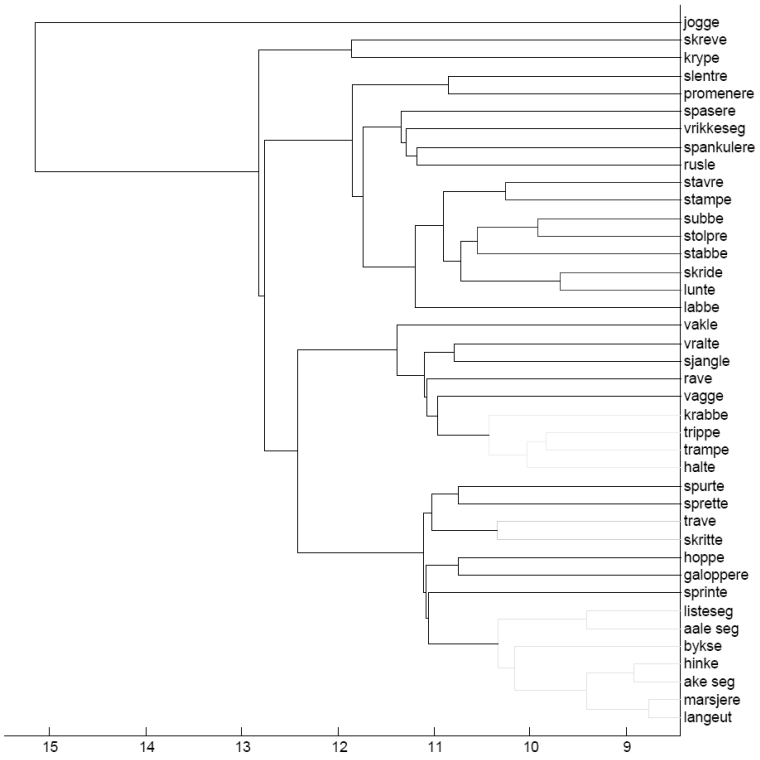


Fig. 5. Clusters based on 15 features, Euclidean metric and Average linkage was used

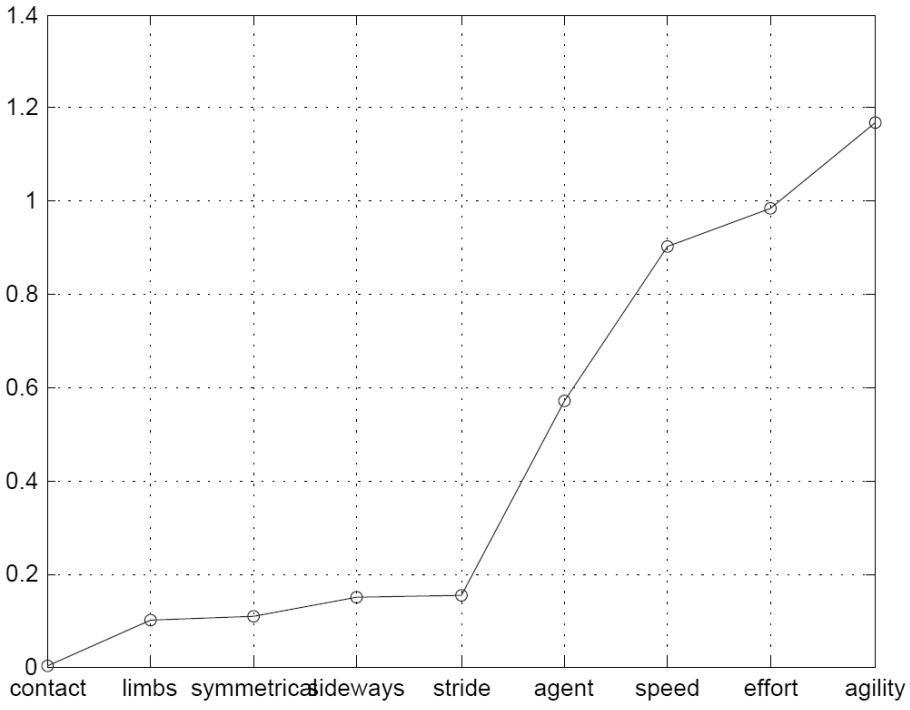


Fig. 6. Weights for 9 features

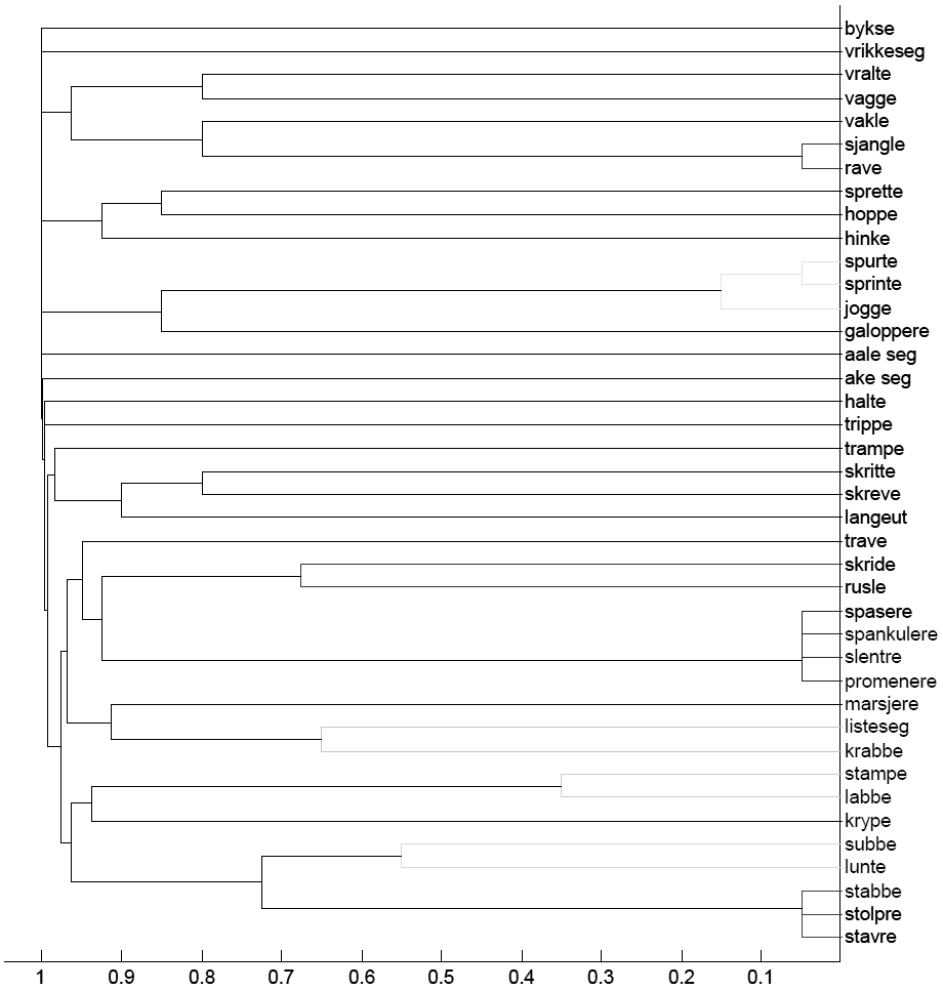


Fig. 7. Clusters based on 9 features, Jaccard metric and Average linkage was used

5. Discussion

The results from the computational method employed have highlighted a number of interesting features of this kind of research. Firstly, they have underscored the validity of combining human data analyses with computational methods. In addition, they have demonstrated that computer modeling of the data can provide useful insights for the underlying semantic similarities, as well as complement or even supplement the human analysis. Concerning the distance and linkage methods, the Euclidian average has proven most useful in representing the underlying similarities in the data, as well as in visualizing the structure of the semantic field of more specific verbs of locomotion. In contrast, the Jaccard distance metric does not seem to capture the structure of the field, and the clusters created by this method appear ad hoc and largely accidental. This is confirmed in our previous work as well, whereby Jaccard plots, while not particularly revealing, were good at capturing subtle details of specific similarities between isolated items.

The method of feature weighting has also proven successful and the removal of features has produced neat and succinct clusters. It is worth mentioning that feature removal has a negative side to it, since it increases the weights of certain features, while removing other features which might be interesting for the analysis. Furthermore, there is a risk of capturing only the overall and more general tendencies in the structure of the semantic field at hand, while missing more subtle aspects of semantic similarity. Our tentative conclusion at this stage is that a set of 9 or 8 features is within the comfortable zone in this respect. The weighted feature cluster with 8 features is most representative of this method and reveals a graded structure of the field of locomotion, with clear-cut clusters defined on a continuum from low-speed, heavy (longer stride), non-

agile motion patterns to high-velocity, agile and effort-demanding locomotions. The middle clusters reflect the importance of contact with the substrate, limb alternation, which are features carrying less weight in the 8-feature plot.

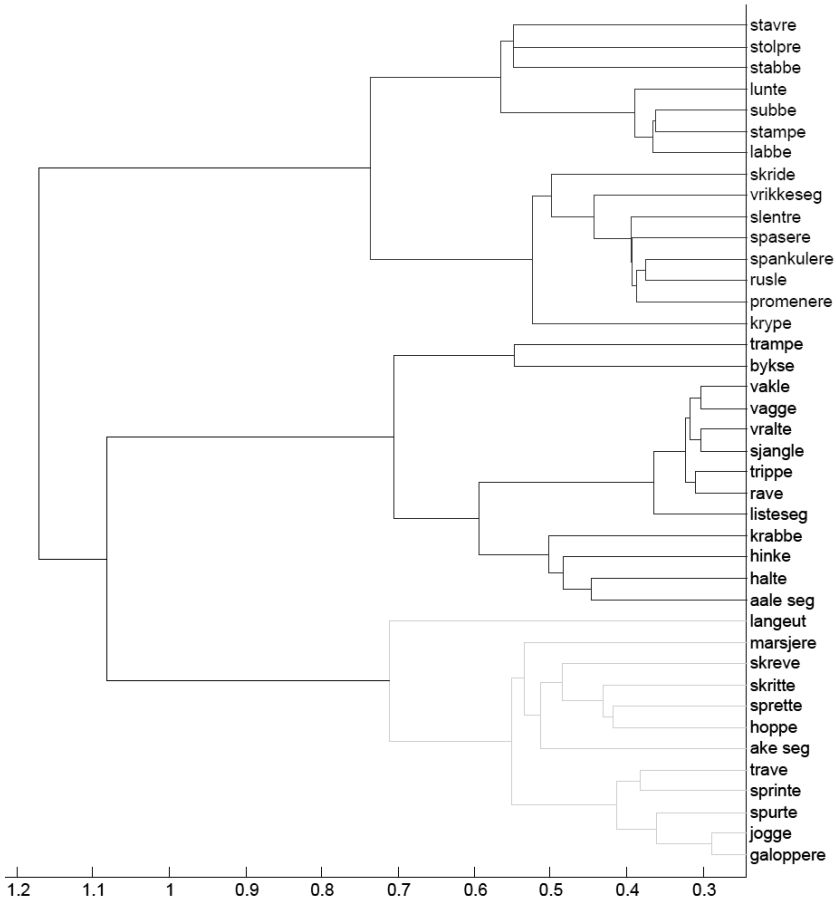


Fig. 8. Clusters based on 9 features, Correlation metric and Average linkage was used

Even though there is no exact match between the cluster obtained from the human sorting data C_{human} and the feature-weighted cluster C_{comp}, they reveal the most salient semantic features relevant for the grouping, such as *speed*, *effort*, *agility*, *contact* with the substrate. We also hypothesise, based on these results that the cluster based on the human data, reflects the individual differences and variation in what features individual speakers find most relevant for the grouping. We further hypothesise that these features are perceptual in nature and may vary according to the specific contexts in which these lexical items were acquired. For instance, for verbs that denote unsteady/swinging gaits, other factors (e.g., speed or effort) may be found irrelevant. In contrast, the cluster obtained by computer modeling and feature-weighting is based on features that the participants mentioned in the subsequent interview session and dictionary definitions of the verbs, and as such are the result of deliberate conceptualization. This finding is interesting in its own right and confirms usage-based accounts of language acquisition as tightly temporally and spatially-bound ([11], [12], [13], [14]).

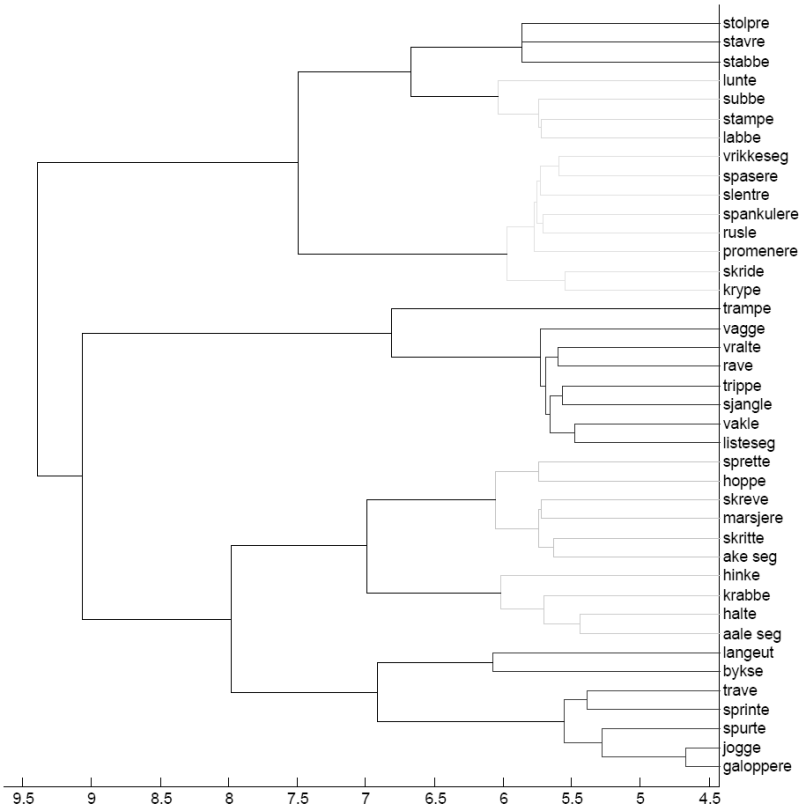


Fig. 9. Clusters based on 9 features, Euclidean metric and Average linkage was used

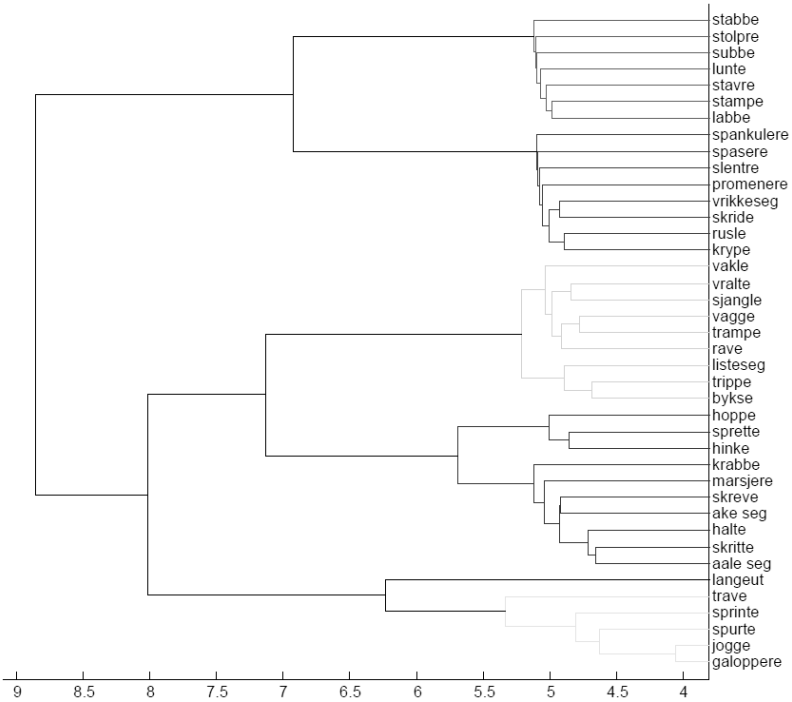


Fig. 10. Clusters based on 8 features, Euclidean metric and Average linkage is used

It is worth noticing that the feature-weighted clusters based on fewer features (8 and 9) still display some anomalies. For instance, verbs like *kryppe* (creep), *krabbe* (crawl for human infants), *ake seg* (move butt-scooting) and *aale seg* (slither, creep like a snake) all belong in different and not immediately coherent clusters, while in the human data cluster they appear on the same branch. What these verbs share, and what is reflected in the human sorting, is the fact that all of these types of locomotion are non-default (for humans), presuppose greater contact with the substrate, in the case of *aale seg*, full body contact with the ground, and the use of more limbs than just the legs. We propose that the feature weighted cluster does not reflect this similarity properly as the result of removing some of the features that underlie the similarity among the above verbs.

6. Conclusion

The results from the computational method employed have highlighted a number of interesting features of this kind of research. Firstly, they have underscored the validity of combining human data analyses with computational methods. In addition, they have demonstrated that computer modeling of the data can provide useful insights for the underlying semantic similarities, as well as complement or even supplement the human analysis.

References

[1] Zhang, B. and Srihari, S.N.: Binary vector dissimilarity measures for handwriting identification. Proc. of SPIE Vol 5010, pp 28-38. (2003)

[2] Debnath, S. and Ganguly, N. and Mitra, P.: Feature weighting in content based recommendation system using social network analysis, *in:* WWW '08: Proceeding of the 17th international conference on World Wide Web. pp. 1041-1042. ACM. New York, NY, USA (2008)

[3] Cordingley, E. S.:Knowledge elicitation techniques for knowledge-based systems, *in:* D. Diaper, Ed. Knowledge Elicitation: Principles, Techniques, and Applications. pp. 89-175. New York: John Wiley and Sons, (1989).

[4] Coventry, K., M. Vulchanova, T. Cadierno, L. Martinez, and R. Pajusalu, in preparation, Locomotion below the basic level: Sorting verbs across languages.

[5] Geiwitz, J. and Kornell, J. and McCloskey, B. P.: An Expert System for the Selection of Knowledge Acquisition Techniques. Technical Report 785-2. Santa Barbara, CA: Anacapa Sciences, (1990).

[6] Roberson, D., Davies, I.R.L., Corbett, .G. and Vandervyver, M. (2005). Free-sorting of colors across cultures: Are there universal grounds for grouping? To appear in: Journal of Cognition and Culture.

[7] Dabrowska, Ewa (2009) Words as constructions. In Vyvyan Evans and Stphanie Pourcel, eds., New Directions in Cognitive Linguistics. John Benjamins, Amsterdam, pp. 201-223.

[8] Rosch, E., C.B. Mervis, W.D. Gray, D.M. Johnson, and P. Boyes-Braem (1976). Basic objects in natural categories. Cognitive psychology, 8, 382-439

[9] Vulchanova, M., L. Martinez and O. Edsberg, in press. A basic level category for the encoding of biological motion, *in:* Hudson, J., C. Paradis and U. Magnusson (eds.) "Conceptual Spaces and the Construal of Spatial Meaning. Empirical evidence from human communication", Oxford: Oxford University Press.

[10] Zachary, W. W. and Ryder, J. M. and Purcell, J. A.: A computer based tool to support mental modeling for human-computer interface design. CHI Systems Tech Report No 900831-8908 (1990).

[11] Smith, L. 2005. Action alters shape categories, Cognitive Science 29, 665679

[12] Coventry, K.R. and Garrod, S. (2004) Saying, Seeing and Acting: The Psychological Semantics of Spatial Prepositions. Hove: Psychology Press

[13] Abbot-Smith, K., and Tomasello, M. (2006). Exemplar-learning and schematization in a usage-based account of syntactic acquisition. Linguistic Review, 23(3), 275-290

[14] Dabrowska, Ewa. (2010). The mean lean grammar machine meets the human mind: Empirical investigations of the mental status of rules, *in:* H.-J. Schmid and S. Handl (eds.), Cognitive foundations of linguistic usage patterns, Empirical approaches (pp. 151-170). Berlin: Mouton de Gruyter.

CLITIC DOUBLING IN BULGARIAN: BETWEEN OPTIONALITY AND OBLIGATORINESS

Teodora Radeva-Bork*

Department of Linguistics, University of Vienna
t.radevabork@gmail.com

ABSTRACT

This paper explores the interaction of the range of possible word orders and direct object clitic doubling in Bulgarian, discussing contexts of optional vs. obligatory clitic doubling. In my analysis, all direct (and indirect) objects that cooccur with pronominal clitics are instances of CD, regardless of the position of the objects in relation to the verb (i.e. in the Left Periphery or to the right of the verb). The main goal is to show that syntactic structure (particularly in a syntactically flexible language like Bulgarian) rather than merely predicate types triggers the obligatory vs. optional presence of a doubling clitic. If predicate choice irrespective of the used construction induced obligatory CD (as suggested in Krapova & Cinque 2008), it could be expected that CD appears without exception when used with such predicates. The data in the paper does not confirm this prediction. I also show that CD functions as a marker of objecthood and topicality in sentences, in which the neutral Bulgarian word order (i.e. SVO) is not obeyed.

1. Introduction

The present paper considers the interaction of the range of possible word orders and direct object clitic doubling in Bulgarian, discussing contexts of optional vs. obligatory clitic doubling (henceforth CD). I establish dependence between syntactic structure and use of CD contexts where the doubling clitic functions as means of object and topic identification. Chapter 2 starts with a general presentation of the CD phenomenon, discussing its spread across the Balkan languages and the problems that arise as a consequence of the uneven distribution of the phenomenon. Chapter 3 surveys the factors that guide the obligatory use of CD in Bulgarian and discusses the object/topic marking function of doubling constructions. The main conclusions of the paper are presented in Chapter 4.

2. The phenomenon of clitic doubling

Clitic doubling refers to the overt doubling of a verbal argument (i.e. the associate) by a weak pronoun – a clitic, inside the same clausal domain. The clitic bears the same phi-features and case as the associate. Further on, the associate can be a full pronoun, a non-pronominal referring expression (DP), a CP, or a wh-word.[1] Both direct and indirect objects can be doubled in this way as shown in (1) and (2).

(1) Mečkata ja xvana Borko.
bear$_{DEF}$ her$_{CL}$ caught Borko
'Borko caught the bear.'

(2) Poštadžijata mu dostavi pismoto na Ivan s goljamo zakăsnenie.
postman$_{DEF}$ him$_{CL}$ delivered letter$_{DEF}$ to Ivan with great delay
'The postman delivered the letter to Ivan with a great delay.'

2.1. Spread of clitic doubling across the Balkan languages

Although CD is perhaps one of the most salient features of the Balkan Sprachbund, it is displayed in varying degrees and is governed by different conditions across the Balkan languages.[2] Whereas it is obligatory with all specific indirect objects and definite direct objects in Macedonian, in Albanian only indirect DPs must be clitic-doubled, and in

* Recipient of a DOC-fellowship of the Austrian Academy of Sciences at the Institute of Linguistics.

[1] Here I restrict myself to the analysis of CD constructions with a full DP associate.

[2] Cf. Dimitrova-Vulchanova & Hellan (1999), Tomić (1996, 2008), Anagnostopoulou (1999), Kallulli (2000), Tasmowski (1987), Franks & King (2000), Rudin (1997), Guentchéva (1994), Friedman (2008) a.o.

Romanian it is the specific indirect objects and topicalized, specific, human direct objects that induce CD. Although, strictly speaking, CD is optional with specific direct and indirect objects in Bulgarian, the lack of clitic doubling often compromises the felicity and even the grammaticality of the utterance and thus makes it obligatory in some cases (cf. Chapter 3.1. for discussion).

2.2. The "genuineness" problem

Probably because CD is unevenly distributed across the Balkan languages and the conditions on its distribution in obligatory vs. optional contexts vary greatly, the analysis of the true nature of CD constructions has yielded a number of controversies. Thus, constructions that typically are considered to show CD, i.e. cases when direct and indirect objects moved to the left periphery invariably trigger the surfacing of a doubling clitics (cf. Dimitrova-Vulchanova & Hellan (1999) for Bulgarian and Alexopoulou & Kolliakou (2002) for Greek), have been analysed inspired by the Romance tradition (cf. Cinque 1984, 1990) as Clitic Left Dislocation (CLLD).[3] CLLD is seen to represent a distinct construction type, different from "true" CD. Other constructions such as Clitic Right Dislocation (CLRD), Hanging Topic and Focus Movement have been introduced to describe constructions types that share some properties with CLLD and CD but yet represent different constructions (cf. e.g., Krapova & Cinque 2008 a.o.).

So the question is what constitutes genuine CD and what factors guide the distribution of the phenomenon?

In Krapova & Cinque (2008), CD is restricted to clauses with certain types of predicates (e.g. psych and physical perception predicates, modal predicates, predicates with possessor datives, etc.). Syntactic construction is seen to be irrelevant and it is rather the relation between types of predicates and the obligatoriness of the doubling clitic that distinguish CD constructions. Under this analysis only (3) but not (4), (5) exemplifies genuine CD.

(3) Ivan *(go) boli kraka.
Ivan him_{CL} hurts leg_{DEF}
'Ivan's leg hurts.'

(4) Majmunite ot filma gi sănuvam vsjaka večer.
$apes_{DEF}$ from $movie_{DEF}$ $them_{CL}$ dream every night
'I dream of the apes from the movie every night.'

(5) Sănuvam gi vsjaka večer majmunite ot filma.
dream $them_{CL}$ every night $apes_{DEF}$ from $movie_{DEF}$
'I dream of the apes from the movie every night.'

In my analysis, all direct and indirect objects that cooccur with pronominal clitics are instances of CD, regardless of the position of the objects in relation to the verb[4] (i.e. in the Left Periphery or to the right of the verb).[5]The fact that CD in languages such as Bulgarian and Macedonian is not dependent on the position of the object in the clause supports the analysis. Furthermore, if there does exist some relation between obligatoriness of the doubling clitic and the appearance of CD, in my analysis the source of this obligatoriness is the syntactic structure rather than the predicate itself (cf. Chapter 3.1. for further discussion). Therefore, I consider all the instances in (3) to (5) above to represent a unitary phenomenon, i.e. genuine CD.

3. Clitic doubling in Bulgarian

This chapter explores the conditions that make certain contexts for CD obligatory in Bulgarian establishing a relationship between choice of syntactic structure and CD, where CD functions as means of topic and object identification.

[3] Cf. e.g., Iatridou (1990), Anagnostopoulou (1994), Arnaudova (2003).
[4] As well as to the clitic since Bulgarian clitics are verbal.
[5] This analysis is in line with Assenova (2002), Guentchéva (1994), Leafgren (1997), Franks & Rudin (2005) for Bulgarian and Tomić (2008) for Macedonian.

3.1. Between optionality and obligatoriness

CD is rare in formal and written Bulgarian. As a consequence of a strong prescriptive tradition, some speakers avoid the use of doubling constructions even in colloquial speech[6] and often it seems that doubling is a purely optional phenomenon since the absence of a doubling clitic does not always lead to ungrammaticality. However, there are certain situations in which CD is obligatory in Bulgarian[7] (cf. Franks & Rudin 2004, Jaeger & Gerassimova 2002, Jaeger 2003 a.o.): (I) when the associate is an oblique subject, as in (6); (II) when it is a topic, as in (7); and (III) when wh-movement appears to violate Superiority, as in (8).

(6) Ivan *(go) sărbi răkata.
Ivan him_{CL} itches arm_{DEF}
'Ivan's arm is itching.'

(7) Marija nikoj ne *(ja) običa.
Maria nobody not her_{CL} loves
'Nobody loves Maria.'

(8) Kogo koj *(go) natupa?
whom who him_{CL} beat
'Who beat whom?'

Bulgarian is characterised by great syntactic flexibility and structure information-driven word order despite a lack of a case marking system. Clitic doubling and the range of possible word orders in Bulgarian are often dependent on each other, and indeed there are cases when CD licenses certain word orders:

(9) Knigite *(gi) izgori Marija.
$books_{DEF}$ $them_{CL}$ burnt Maria
'Maria burnt the books.'

(10) Izgori *(gi) Marija knigite.
burnt $them_{CL}$ Maria $books_{DEF}$
'Maria burnt the books.'

In fact, as it has been previously discussed (cf. Rudin 1986, Werkmann 2003), if the preferred S-V-DO-IO surface order is not followed, CD is necessary to identify the syntactic roles of object vs. subject. If this is not done, the correct interpretations for (11) and (12) are grammatically excluded since if not doubled, the fronted objects *dvete nevinni žertvi* and *Boris* will be wrongly interpreted as subjects:

(11) Dvete nevinni žertvi *(gi) izjali vălzi tazi sutrin.
two_{DEF} innocent victims $them_{CL}$ ate wolves this morning
'The two innocent victims were eaten by wolves this morning.'

(12) Boris izvednăž *(go) svali bolestta na legloto.
Boris suddenly him_{CL} knock down $sickness_{DEF}$ onto bed_{DEF}
'Boris was knocked down by a sudden sickness.'

[6] The elicitation of CD contexts in Bulgarian may prove difficult as indicated in Jaeger & Gerassimova (2002: footnote 6). Leafgren (2002) provides a valuable source of analysing a corpus of Bulgarian colloquial data that allows making judgements based on actual usage rather than on speakers' judgements of what they think they say.

[7] I concentrate only on the first two situations.

In (11) the semantic status of the DP associate plays no role with regards to the obligatoriness of CD: the object must be doubled no matter if it is definite or indefinite. This is predicted by the present analysis since the construction itself triggers the presence of the clitic.

A possible explanation for the difference between the constructions in (11), (12), on the one hand, and (3), on the other hand, can be that in (11) and (12) the doubling clitic plays a role in the syntax, whereas in (3), the clitic is part of the lexical item (i.e. impersonal verb+clitic).[8] In other words, in one case the clitic is just part of the lexical entry like in *boli me* "it hurts me" whereas in the other case its use is necessitated by the syntax, (i.e. type of construction used) in order to identify syntactic roles, and often to resolve ambiguity. Evidence for this is supplied by the existence of pairs of predicates with and without a clitic, e.g. *haresva mi* "it appeals to me" and *haresva*; *boli me* "it hurts me" and *boli*; *spi mi se* "I feel like sleeping" and *spi*, etc. This difference can explain the obligatoriness (but only in some cases) of CD with this special subset of predicates.

Recall that Krapova & Cinque's (2008) analysis of CD proper is based on the idea of a relation of certain types of predicates and obligatoriness of a doubling construction (cf. Chapter 2.2.). A closer inspection shows that not all of the listed predicates induce obligatory CD *irrespective of the construction used*. If oblique subjects co-occur with a nominative argument, CD is not obligatory (cf. (13a) vs. (13b), (14a) vs. (14b), and (15a) vs. (15b)). In (13b-15b) CD is not necessary as there is no mismatch between the syntactic positions of the arguments in the neutral SVO order and their syntactic roles.

(13) a. Omrăzna *(i) da gleda televizia (na Marija).
got tired her$_{CL}$ to watch TV (to Maria)
'Maria got tired of watching TV.'

b. Televiziata/Gledaneto na televizia (i) omrăzna bărzo na Marija.
TV$_{DEF}$/ watching$_{DEF}$ of TV her$_{CL}$ got tired quickly to Maria
'TV/Watching TV quickly got Maria tired.'

(14) a. V poslednia moment *(mu) xrumna, če e zabravil da izkluči utijata.
in last$_{DEF}$ moment him$_{CL}$ occurred that is forgot to switch-off iron$_{DEF}$
'It occurred to him in the last minute that he had forgotten to switch off the iron.'

b. (Tova) ce e zabravil da izkluči utijata (mu) xrumna na Ivan v poslednia moment.
(this) that is forgot to switch-off iron$_{DEF}$ him$_{CL}$ occurred to Ivan in last$_{DEF}$ moment
'That he had forgotten to switch off the iron occurred to Ivan in the last moment.'

(15) a. Na Ivan *(mu) dosažda pesenta.
to Ivan him$_{CL}$ bothers song$_{DEF}$
'Ivan is bothered by the song.'

b. Pesenta (mu) dosažda na Ivan.
song$_{DEF}$ him$_{CL}$ bothers to Ivan
'The song bothers Ivan.'

Additionally, the data in (16) show that the predicates which are expected to induce CD all over, show a different behaviour when the associate is omitted as CD does not take place.

[8] Guentchéva (2008) similarly mentions that the accusative or dative clitic in impersonal constructions is "an integral component of the predicate".

(16) a. Glavata boli mnogo pri padane ot visoko.
head$_{DEF}$ hurts much after falling from high
'The head hurts badly after falling down from a high place.'

b. Ušite mnogo boliat (pri văzpalenie).
ears$_{DEF}$ much hurt (at inflammation)
'The ears hurt badly (when inflamed).'

The sentences in (16) make available a different semantic interpretation than the one in the equivalents with a clitic-doubled associate, apparently due to semantic conditions on the used DPs. This paper cannot include a discussion of the semantic factors at play here, but what is important for the analysis is that the predicate itself cannot be the only defining factor for use vs. non-use of CD.

3.2. Topicality and objecthood

Since Bulgarian allows for objects to occur sentence initially, CD is necessary for the identification of syntactic and information structure if the object precedes the subject. In other words CD is the only means of signaling objecthood and topicality in such cases.[9] Even very young children at the age of 3 to 4 years seem to be sensitive to the object/topic identification function of CD constructions since they successfully identify fronted clitic-doubled arguments as the syntactic objects in sentences similar to (4), (11), (12) above (cf. Radeva-Bork, in progress). These findings are in line with the results of Jaeger & Gerassimova's (2002) online study showing that fronted, topical objects are always doubled as well as Dimitrova-Vulchanova & Vulchanov's (2008) data from Old Bulgarian showing that contrastive topic is a trigger for the surfacing of doubling clitics. Leafgren (2002) also demonstrates that object reduplication in Bulgarian is almost always used as an overt marker of topicality.[10]

Object marking in CD constructions is particularly important when reciprocal verbs are used. In such cases the use of CD becomes obligatory (cf. (17a) vs. (17b)).

(17) a. Marija nikoj ne celuna.
Maria nobody not kissed
'Maria kissed nobody.'

b. Marija nikoj ne *(ja) celuna.
Maria nobody not her$_{CL}$ kissed
'Nobody kissed Maria.'

CD is optional in constructions with the neutral SVO word order. In (17b), it is the particular word order, i.e. SUBJ not first, that triggers the obligatory use of CD as means of a disambiguation between *Marija* as a subject as in (17a) vs. object as in (17b).

4. Conclusion

Based on the data presented in this paper, I suggest that CD in Bulgarian cannot be reduced solely to cases of CD in obligatory contexts, e.g. with certain psych and physical perception predicates. In Bulgarian, a syntactically flexible language with a structure information-driven word order, choice of syntactic structure rather than only predicate choice is the driving factor with regards to whether clitic doubling is optional or obligatory.
Bulgarian allows its objects to occupy different syntactic positions. The presence of a doubling clitic is often means of object and topic identification in utterances that do not conform to the neutral SVO order. In turn, CD in Bulgarian is not dependent on the position of the doubled object with regards to the verb. I suggest that constructions such as in (3), (4) and (5) (cf. Chapter 2.2.), all present true cases of CD.

[9] Apart from intonation.
[10] The idea of obligatory CD for topic-fronted objects has also been discussed in Alexandrova (1997), Dimitrova-Vulchanova & Hellan (1995/99), Leafgren (1997), Rudin (1997), Jaeger & Gerassimova (2002), Jaeger (2003), Guentchéva (2008).

References

Alexandrova, G. 1997. Pronominal clitics as g(eneralized) f(amiliarity)-licensing agro. In Browne et al. (eds.). *Formal Approaches to Slavic Linguistics: The Cornell Meeting*, 1-31. Ann Arbor: Michigan Slavic Publications.

Alexopoulou, T., Kolliakou, D. 2002. On linkhood, topicalization and clitic left dislocation. *Journal of Linguistics*, 38: 193-245.

Anagnostopoulou, E. 1994. *Clitic Dependences in Modern Greek*. Ph.D. dissertation, Universität Salzburg.

Anagnostopoulou, E. 1999. On the representation of clitic doubling in Modern Greek. In van Riemsdijk, H. (ed.). *Clitics in the Languages of Europe*, 761-798. Berlin: Mouton de Gruyter.

Arnaudova, O. 2003. *Focus and Bulgarian Clause Structure*. Ph.D. dissertation, University of Ottawa.

Assenova, P. 2002. *Balkansko ezikoznanie. Osnovni problemi na Balkanskija ezikov săjuz*. Sofia: Faber.

Cinque, G. 1984. Clitic Left Dislocation in Italian and the 'Move-α' parameter. MS. Universitá di Venezia.

Cinque, G. 1990. *Types of A'-dependences*. Cambridge: MIT.

Dimitrova-Vulchanova, M., Hellan L. 1995/99. Clitics and Bulgarian clause structure. In van Riemsdijk, H. (ed.). *Clitics in the Languages of Europe*, 469-514. Berlin: Mouton de Gruyter.

Dimitrova-Vulchanova, M., Hellan L. (eds.). 1999. *Topics in South Slavic Syntax and Semantics*. Amsetrdam: John Benjamins.

Dimitrova-Vulchanova, M., V. Vulchanov. 2008. Clitic doubling and Old Bulgarian. In: Kallulli, D., L. Tasmowski (eds.). *Clitic doubling in the Balkan languages*, 105-132. Amsterdam: John Benjamins.

Franks, S., C. Rudin 2004. Bulgarian clitics as K^0 heads. Presented at FASL 13, South Carolina.

Franks, S., C. Rudin. 2005. What makes clitic doubling obligatory. MS.

Franks, S., T.H. King. 2000. *A Handbook of Slavic Clitics*. Oxford: OUP.

Friedman, V.A. 2008. Balkan object reduplication in areal and dialectological perspective. In: Kallulli, D., L. Tasmowski (eds.). *Clitic doubling in the Balkan languages*, 35-63. Amsterdam: John Benjamins.

Guentchéva, Z. 1994. *Thématisation de l'objet en bulgare*. Bern: Peter Lang.

Guentchéva, Z. 2008. Object clitic doubling constructions and topicality in Bulgarian. In: Kallulli, D., L. Tasmowski (eds.). *Clitic doubling in the Balkan languages*, 203-223. Amsterdam: John Benjamins.

Iatridou, S. 1990. Clitics and island effects. *UPenn Working Papers in Linguistics*, 2: 11-38.

Jaeger, F. 2003. Topicality and superiority in Bulgarian wh-questions. Presented at FASL 12, Ottawa.

Jaeger, F., V.A. Gerassimova. 2002. Bulgarian word order and the role of the direct object clitic in LFG. In Butt, M., T.H. King (eds.). *Proceedings of the LFG Conference 2002*, 197-219. Stanford: CSLI Publications.

Kallulli, D. 2000. Direct object clitic doubling in Albanian and Greek. In Beukema, F., M. den Dikken (eds.). *Clitic Phenomena in European Languages*, 209-248. Amsterdam: John Benjamins.

Krapova, I., G. Cinque. 2008. Clitic reduplication constructions in Bulgarian. In: Kallulli, D., L. Tasmowski (eds.). *Clitic doubling in the Balkan languages*, 257-287. Amsterdam: John Benjamins.

Leafgren, J. 1997. Bulgarian clitic doubling: Overt topicality. *Journal of Slavic Linguistics*, 5: 117-143.

Leafgren, J. 2002. *Degrees of Explicitness*. Amsterdam: John Benjamins.

Radeva-Bork, T. in progress. An elicitation study on the comprehension of clitic doubling constructions with Bulgarian monolingual children (3-4 years). University of Vienna.

Rudin, C. 1986. *Aspects of Bulgarian syntax: Complementizers and WH constructions*. Columbus: Slavica.

Rudin, C. 1997. AgrO and Bulgarian pronominal clitics. In *Formal Approaches to Slavic Linguistics: The Indiana meeting* 1996, 224-252. Ann Arbor: Michigan Slavic Publications.

Tasmowski, L. 1987. La reduplication clitique en roumain. In Plangg, G., M. Iliescu (eds.). *Rätoromanisch und Rumänisch. Akten der Theodor Gartner-Tagung*, 377-399. Innsbruck: Amae.

Tomić, O. Mišeska. 1996. The Balkan Slavic clausal clitics. *Natural Language and Linguistic Theory*, 14: 811-872.

Tomić, O. Mišeska. 2008. Towards grammaticalization of clitic doubling. In: Kallulli, D., L. Tasmowski (eds.). *Clitic doubling in the Balkan languages*, 65-87. Amsterdam: John Benjamins.

Werkmann, V. 2003. Objektklitika im Bulgarischen (= studia grammatica 57). Berlin: Akademie Verlag.

ON THE TROCHAIC FEET, EXTRAMETRICALITY AND SHORTENING RULES IN STANDARD SERBIAN

Stanimir Rakić

non-affiliated

Blv. Arsenija Čarnojevića 37, 11070 Belgrade, Serbia

starakic@gmail.com

Abstract

In this paper I try to show that shortening rules in standard Se rbian can be intrpreted as trochaic shortenings. Such an interpretation would not be possible if we accept the claim of Zec (1999) that all accented syllables in Neoštokavian are, in fact, heavy.

1. Introduction

My main thesis in this paper is that a number of shortening rules in standard Serbian (henceforth SS) can be interpreted as trochaic shortenings.[1] I agree with Zec (1999) that the dis tribution of stress in SS is "largely predictable", and try to back up this claim by giving a provisional rule for noun accent in the next section. Zec (1999) also claims that, in Neoštokavian, every syllable associated with tone must be granted foot status. However, the shortening rules strongly suggest that this claim is wrong in its general, unrestricted form. If her thesis were true, neither of shortenings described in this article would make sense, because only two types of feet would be possible – $(H)_F$ and $(HL)_F$, where H denotes a heavy syllable, and L a light one. In this paper we assume the classical inventory of feet as defined in Halle & Vergnaud (1987), Hayes (1995) and Kager (1989), and, in particular, the following hierarchy of Prince (1990):

(1)	iamb:	(LH)	>	(LL),(H)	>	(L)
	trochee:	(LL),(H)	>	(HL)	>	(L)

On the basis of this hierarchy we expect the elimin ation of monomoraic foot $(L)_F$ and, in trochaic systems, the shortening of the foot (HL) into the optimal one (LL). If we accept the suggestion of Zec that every accented syllable in Neoštokavian is heavy , no trochaic shortening would be possible as there would be no foot of the type $(LL)_F$. All accents are checked in *Rečnik srpskoga jezika* (further RSJ) and *Rečnik srpskohrvatskoga književnog jezika* (further RSKJ).

2. The Distribution of Accents and E xtrametricality

According to the traditional view the risi ng accents in SS may take any position in the word except the last one. It is not difficult to show that the position of rising accents in monomorphemic nouns is generally restricted to the penult and antepenult position; the accent further left than the a ntepenult is the result of affixation and compounding. In view of accents , one can distinguish five main kinds of suffixes in SS: cyclic, receptive, extracyclic, extrametrical and dominant (Ракић 1991a, Rakić 1991b). The cyclic, extracyclic and extrametrical suffixes perform the same function as in English, while the so called "receptive" suffixes have a special property to change the accents of derivatives only if the last syllable of a ste m contains an unaccented length.

[1] What I have to say holds equally in all languages whose standard is based on the Neoštokavian dialect, i.e. in Croatian and in newly established Bosniak.

As in English, the main accent in nouns may fall on the penult or on the antepenult, but the penult is preferred if the final syllable is heavy; if the final syllable and the penult are light, the antepenult is preferred in trisyllabic words, but in bisyllabic words the penult is accented. For the notion of extrametricality one can stipulate that the final heavy syllable in Serbian is counted as light, and that the final light syllable as extrametrical. It seems, however, that extramerticality in SS is largely lexically determined, especially in foreign borrowings.

We start with the assumption that the foot system in SS consists of standard trochaic feet of one heavy syllable or two light syllables. As in many other languages, light feet are generally avoided or eliminated. For example, in some Serbian and Croatian dialects, we come across the so-called Kanovian lengthening in which the first syllable in the nouns like vòda 'water' and sèlo 'village' is lengthened (i.e. vòda > vóda, *Ивић* 1985: 75, Hraste 1957). In SS, however, the other possibility is adopted: in the bisyllabic nouns like vòda extramtericality is eliminated and these words are pronounced as a foot with two light syllables (e.g. vòda, sèlo).

The presence of trochaic foot usually implies the succession of secondary stresses in alternative syllables. In SS, secondary accents are not expressed with much strength, but Jokanović - Mihajlov (2008) confirms their existence in longer words and slower speech (e.g. prèpo-rùčujēm vam 'I recommend you', nèpri-znâte zasluge 'unrecognized merits'). As we could expect, the binary feet prevail in her analysis. Similarly, Belić (1948: 108) observes that all unaccented length following the accent can be interpreted as longfalling secondary accents. Applying these observations, we can supply footing to trisyllabic and tetrasyllabic nouns with length in the last syllable, e.g. đìgitrōn = (đìgi)(trōn).

3. The Shortening Rule

The definition of extrametricality given above seems to be required by the following shortening rule:

(2) The length in the stem is shortened before polysyllabic suffixes or suffixes which consist of one closed syllable.

The examples (3) illustrate the application of this rule:

(3) glâs 'voice'< glàsāč 'voter', glûp 'stupid'< glùpān 'dumb person', bômba 'bomb'< bòmbāš 'bombardier', crêp 'tile'< crèpara 'tile factory' , màršāl 'marshal' < maršàlāt 'marshal's office', sôm 'sheatfish'< sòmina aug., mètōd 'method' <metòdika 'teaching methods', gûst 'dense'< gùstiš 'bush', dìvljāk 'savage'<divljàkuša fem., etc. (Rakić 1996a, 1996b),

The rule (2) is a lexical rule which applies effectively before all cyclic suffixes and most receptive suffixes. The rule (2) can be understood as trochaic shortening only if we assume extrametricality as defined above. The extrametricality marking reduces the suffixes *-āč, -ān, -āš, -āt* and *-iš* to light syllables, so that trochaic shortening (HL) □ (LL) can be applied. At the same time extrametricality explains why there is no shortening before the receptive suffixes containing just one light syllable:

(4) -če (dućánče 'shop'), -stvo (pápstvo 'pope's authority'), -će (otkŕíće 'discovery').

The closed suffixes *-ov (bânov* 'the ban's') and *-in (bébin* 'the baby's'*)* fulfill the condition for shortening, but with the derived possessive adjectives the paradigmatic identity is of paramount importance. The paradigmatic identity also prevails with *-ōst* deriving abstract nouns from adjectives (e.g. *vulgárnōst* 'vulgarity', *labîlnōst* 'lability') and with *-(j)anin* deriving the names of

inhabitants of towns and regions (e.g. *Brúšanin* 'an inhabitant of Brus'). The length is preserved mainly in nouns containing prefixes (*zázor* 'inhibition', *rásad* 'nursery plant', *príroda* 'nature').The main reason for the preservation of length in these nouns is that the rule (2) applies only locally, and prefixes normally are not adjacent to suffixes. (2) also does not apply to foreign borrowings, which are usually adopted as a whole (*ášov* 'shovel'*, báger* 'dredge'*, bárut* 'gunpowder' *, dôboš* 'drum'*, mâjstor* 'master'*,* ect.) nor to the nouns which could not be considered derivatives because they have acquired special meanings (e.g. *vûkovac, Kârlovac, Vînkovci,* etc.).

4. The Shortenings in Paradigms and C ompounds

Already Daničić (1925: 32-34, 39, 43) noted the relevant examples of shortening in the declention of masculine nouns with the unstable *a*:

(5) tàlac 'hostage' - tàoca (<tâlca) gen.sg., žètelac 'reaper'- žèteoca (< žètēlca) gen.sg., vládalac 'ruler'- vládaoca (< vládālca) gen.sg., záselak 'hamlet' - záseoka < (zásēlka) gen.sg.

Similar shortenings we can find also in some nominal doublets with the alternation /l/ > 0 (e.g. *žálce* 'snake's tongue'- žàoce, *rílce* 'snout' - řioce, s. RSKJ). Here, the alternation /l/ > /o/ is optional. The same type of shortening is found in the declension of neutral nouns with extended stems in oblique cases (*déte* 'child' - *dèteta* gen.sg.. *párče* 'piece' - *pàrčeta* gen.sg.), and in that of masculine nouns which have optional plural extension *-ov* (*drûg* 'comrade' - *drùgovi* pl., but *drûzi* pl., *vïtēz* 'knight' - *vïtezovi* pl., but *vïtēzi* pl.). However, some monosyllabic nouns preserve the length of their stems (*vâl* 'wave'- *válovi* pl.) presumably under the pressure of faithfulness principle. The change of melody logically fo llows from the assumption that *-ov* is a receptive suffix. In some foreign borrowings we find the shortening of the final syllable before the genitive ending *-a* (*kàfē* 'café' - *kafèa* gen.sg.*, atèljē* 'studio'- *ateljèa* gen.sg. It is interesting that the shortening in (5) requires the extrametricality of the last light syllable (there is no shortening in *tâlca),* but disregards it in the borrowings like *kafèa.* Extrametricality in Serbian seems to be largely lexically determined, especially in foreign borrowings.

In SS, in trisyllabic compounds with falling accent on the first syllable, the first syllable is always short. We can show that th is generalization is just a consequence of the trochaic shortening. We assume provisionally that in words with falling accents in SS, footing procedes from left to right, with the foot at the left end of the word detemining the main accent. For example, in the compound *glùvonēm,* we get the division (glūvo)(nēm). The first foot has the structure (HL), and the application of trochaic shortening provides the structure (gluvo) $_F$(nēm)$_F$. The result is therefore the compound *glùvonēm.* Such shortenings happen regularly if the first component has just two syllables since the linking *o* has no length. It is important that in the mutch rarer cases of nominal compounds beginning with rising accents, the first syllable bears a short rising one (*vràpseme*).

5. The 'iambic' shortening

There is another shortening rule , dubbed by Hayes (1995) iambic shortening, which changes the foot with the structure (LH) into one with the structure (LL) since the foot (LH)$_F$ do not fit into a trochaic system:

(6) The length of the syllable is shortened after the short rising accent.

This rule applies in a number of word formation processes.

(7a) grȁđāni ’citizen’, bȓđāni ’highlanders’ vs. čòbani ’shepherd’, hřišćani ’Christians’;

(b) dòdir ‘touch’ <dodírnuti ‘to touch’, pòpis ‘ list’< popísati ‘to list’;

(c) golòbrad ’beardless’< bráda, dugònos ' long -nosed' < nôs, bèskraj ‘infinity’ < krâj;

(d) životòpis ’biography’– písati ’to write’, nogòstup ’tread’ < stúpati ’step on sth.’;

(e) zȁiskāla pp. ’began to seek’- zaïskala pp., prȍčitāla pp. ‘read through’ - pročitala pp.;

In (7a) the length of the suffix *-ān* is shortened after a shortrising accent in the derived names of inhabitants of towns and regions, but not after a short falling one (Ракић 2005). In (7b) the underlying verbal length is shortened in nouns in the process of conversion (Ракић 1999). Similar shortenings we find in the compounding (7c,d), where the length of the second constituent is shortened following the rising accent and a trochaic foot is formed at the word end (Ракић2004). In (7e) the shortening of length emerges after the shortrising accent in accentual doublets of past participles (Јокановић – Михајлов 2009).

If we assume Zec’s claim that every accented syllable in Neoštokavian is heavy, and therefore can form a foot, instead of the input foot (LH) in (7), we would get two feet (H)(H), a structure in which no shortening is possible.

6. The trochaic lengthening

Prince (1990) notes that in trochaic systems a complementary phenomen of lengtheing is used to eliminate undesirable monomoraic foot. In some English dialects, the first syllables in the words like *police* is lenthened to compensate for the extrametricality of the second (eg. *police* /pó:lìs/*, Detroit* /dí:trɔ̀lt/*, cement* /sí:mènt/, *Arab* /é:ræ̀b/). Chung (1983) reports that in Chamorro, in which extrametricality is lexically determined, “the vowels are lengthened if they bear primary stressand and occur in a penultimate open syllables”.[2] The source of such phenomena is binarity – the requirement that the foot consist of two elements.

Kanovian lengthening in the nouns *vóda, žéna, sélo* in some Neoštokavian dialects can be explained in the same way. The example of Kanovian lengthening is not the only example which illustrates the lenthening of open penults in Neoštokavian. The two -syllabic hypocoristic nouns ending in *-a* bear longrising accent on open penult (e.g. *kóka* hypo. ’hen’*, góspa* hipo. ‘lady’ *, úča* hypo. ’teacher’, *sélja* hypo. ’peasant’, etc.). From feminine names we get numerous hypocoristics with long rising accent (*Bósa, Vîda, Désa, Zóra, Ráda,* etc.), and the same holds for masculine hypocoristcs *Pája/o, Gája/o, Léka/o, Nîka/o*). If the first syllable is closed, lengthening is not necessary in hypocoristics (e.g. *Tȓïpko, Stȅpko, Pȅrko, Rȁstko, Vlȁtko*). This shows that closed syllables in Neoštokavian have to be interpreted as heavy if they are not final. This refutes the claim of Zec (1999) that closed syllables are always light in Neoštokavian.

References

Belić 1951-А.Белић. *Савремени српскохрватски књижевни језик*, I део: Гласови и акценат, Београд.

Calabrese, A. 1983/4. Metaphony in Salentino, *Revista di Grammatica Generativa.*

Chung, S. 1983. Transderivational Relationships in Chamorro Phonology, *Language* 59, 35-66.

Daničić 1925. - Ђ.Даничић. *Српски акценти,* Београд – Земун.

[2] Cf. historical diphtongization under stress of lax mid vowels in Italian (*piéde* < *péde(m), miéle* < *méle(m)*, Calabrese 1983/4).

Hraste, M. 1957. O kanovačkom akcentu u Hrvatskoj, *Filologija* 1, 59-75.
Ивић, П. 1985. *Дијалектологија српскохрватског језика,* Нови Сад: Матица српска.
Jokanović-Mihajlov 2008 - J. Јокановић-Михајлов. Речи са два акцента у савременом српском језику, *НССВД* 37/1, 35-42.
Јокановић-Михајлов, J. 2009. Промена акцента радног глаголског придева, *НССВД* 38/1, 43-51.
Ракић, С. 1991a Циклични и неутрални суфикси у српскохрватском језику, *НССВД* 20/2, 417-426.
Rakić, S. 1991b. O receptivnim sufiksima i pravilu akcenta srpskohrvatskog jezika, *Зборник Матице српске за филологију и лингвистику* 34/*2,* 121-134.
Rakić, S. 1996a. Suffixes, lexikalische Schichten und Akzent im Serbokroatischen, *Linguistische Berichte* 163, 227- 252.
Ракић, С. 1996б. Правило крађења у српскохрватском језику, *Зборник Матице српске за филологију и лингвистику* 39/1, 141-156.
Ракић, С. 1999. О нултим суфиксима и извођењу наставцима *–ø. –а, -о/-е, Зборник Матице српске за филологију и лингвистику* 42/1*,* стр. 225-254.
Ракић, С. 2004. О акценту и дужини именичких сложеница, *Зборник Матице српске за филологију и лингвистику* 47/1-2, стр. 425-444.
Ракић, С. 2005. О извођењу имена становника градова и области (*-анин* један суфикс или два), *Зборник Матице српске за филологију и лингвистику* 48/1-2, 267-276.
Ракић, С. 2008. О дистрибуцији дугих акцената у српском језику, *НССВД* 37/1, стр.339-350.
Zec, D. 1999. Footed tones and tonal feet: rhythmic constituency in a pitch-accent language, *Phonology* 16.

FACTORS INFLUENCING THE PERFORMANCE OF SOME METHODS FOR AUTOMATIC IDENTIFICATION OF MULTIWORD EXPRESSIONS IN BULGARIAN

Ivelina Stoyanova

Department of Computational Linguistics

Institute for Bulgarian Language, Bulgarian Academy of Sciences

ABSTRACT

The paper presents an analysis of some factors influencing the performance of automatic identification of multiword expressions in Bulgarian. The methods applied for the analysis include: (1) a combined method for automatic identification of multiword expressions in Bulgarian which uses both statistical tests and linguistic information; and (2) the method of latent semantic analysis.

It is recognized that the multiword expressions comprise a complex set of linguistic entities with a wide variety of characteristics. The paper discusses the approach towards the classification of multiword expressions as one of the factors influencing the choice of methods for their recognition. Another important set of factors are the characteristics of the resources applied in the study – size and domains of corpora and available annotation (POS tagging, word sense annotation, etc.).

Some examples are presented to illustrate the importance of finding a suitable method for any particular purpose taking into account the characteristics of the analysed corpus.

1. Introduction

Statistical analyses show that multiword expressions (MWEs) comprise a significant part of the lexical system of many languages, for instance 24.49% of Bulgarian WordNet, as well as 22.5% of Princeton WordNet 2.0 (Koeva 2005). MWEs pose a complex set of problems to both theoretical linguistics and Natural Language Processing (NLP). Solving the problems of their automatic identification and treatment will help improving the results in areas such as Information Retrieval, Machine Translation, etc.

A wide variety of approaches towards MWE recognition have been developed and tested in recent years. Generally, they differ in the amount of linguistic information used and the particular statistical tools applied in the analysis. However predominantly statistical methods and methods highly dependant on linguistic resources are proven unsuited for the general purpose of MWE recognition and classification and more attention is paid to the combination of both approaches.

The paper presents two independent methods for automatic identification of MWEs and discusses their applicability for different purposes. Factors influencing performance are discussed the most important among which being the parameters of the task and the characteristics of the resources – corpora and annotation.

2. Two methods for automatic identification of multiword expressions

For the purpose of the study two methods were applied to various Bulgarian resources to analyse the influence of factors on their performance. These include: (1) a combined method using statistical tests and linguistic analysis; and (2) the method of latent semantic analysis. These are briefly outlined below.

The following corpora were employed in the study: Brown Corpus of Bulgarian[1] – a general corpus of one million words, and various special domain corpora which are part of the Bulgarian National Corpus[2] (Koeva et al. 2006). For the purposes of the present study we restrict the constructions under observation to noun phrases of the form adjective – noun only.

2.1. Combined method using statistical test and syntactic filter

The first method applies statistical tests to extract collocations with a higher probability of being MWEs, and then a syntactic filter to eliminate invalid constructions. The method is described in Justeson and Katz (1995). It gives relatively good results taking into account its simplicity and the limited resources it requires (only POS tagging as a preprocessing step). However

[1] More information at http://dcl.bas.bg/Corpus/home_en.html
[2] More information at http://www.ibl.bas.bg/en/BGNC_en.htm

this method is most suited for extracting MWEs the parts of which are adjacent and additional processing is required to adapt it for the task of identifying non-adjacent MWEs.

In our application of the method mutual information (MI) is adopted as the quantitative measure for deciding whether the co-occurring words form a collocation (see Manning and Schutze 1999). It is recognized that this measure as well as most of the other statistical measures does not work well for low frequency events so we only consider bigrams occurring 5 times or more in the general Brown Corpus of Bulgarian. The list of all bigrams of the required frequency is ordered by the MI value. Further, the syntactic filter is applied and only entities of the form adjective – noun are observed. The bigrams are then manually classified into categories: MWEs and free phrases (see 3.1) and the results are evaluated.

Koeva (2007) presents a previous application of a variation of the method for multiword term extraction in Bulgarian.

2.2. Latent Semantic Analysis

The second method uses latent semantic analysis (LSA) which is at first applied in the field of text categorization (Deerwester et al. 1990) and its application for identification of multiword expressions is described in Baldwin et al. (2003) and Katz and Giesbrecht (2006). In its framework the meaning of a word is considered to be a vector in *n*-dimensional vector space where each dimension is represented by a meaningful word of general lexis. Further, Singular Value Decomposition (SVD) is applied and the dimension of the vector space is reduced. LSA provides a way to measure the similarity between a MWE and its constituent words using the value of the cosine of the angle between any pair of vectors.

In our application of the method we use a general lexicon of about 3000 words (excluding any closed class words) considered to comprise the core of the lexicon and relatively constant across domains. This ensures that the way of constructing the vector space is suitable for describing words from any domain. The dimensionality of the vector space is reduced to 100 using SVD over the matrix of the most frequent 50 000 words in the general Brown Corpus of Bulgarian. Then using the newly defined basis for the 100-dimensional vector space the similarity values are derived between the entity (bigram) and each of its constituent words.

Baldwin et al. (2003) formulate the hypothesis that if the similarity between the MWE and its constituents is sufficiently high, then the MWE is simple decomposable. Low similarity would mean that the MWE is either a non-decomposable or idiosyncratically decomposable as the meaning of the constituents does not build (fully) the meaning of the MWE. We aim at defining ranges for the similarity values so that the categories of MWEs are possible to distinguish on that basis.

To the best of our knowledge the method was not previously applied for Bulgarian.

3. Some factors influencing the performance of automatic recognition of MWEs

3.1. Parameters of the task performed – classification of MWEs

We adopt the classification of MWEs presented by Baldwin et al. (2003). The authors distinguish between the following three categories: non-decomposable MWEs where a decompositional analysis of the meaning is not possible (e.g. BG *ovcharska torbichka* – EN *shepherd's purse*); idiosyncratically decomposable MWEs where some components of the phrase have a meaning unavailable outside of the MWE (e.g. BG *periodichna tablitsa* – EN *periodic table*); and simple decomposable MWEs the meaning of which can be decomposed to that of their constituents but they comprise a single lexical unit, e.g. by institutionalization often exhibiting restriction on syntactic structure or synonymic substitutions within the unit (e.g. BG *Bulgarian language* – EN *Bulgarian language, Bulgarian*). On the other hand we have free (not-connected) phrases which are decomposable and are not considered a lexical unit (e.g. BG *vazhen faktor* – EN *important factor*).

In some cases we may be interested in simply distinguishing between MWEs and free phrases in order to define separate methodologies for their treatment, e.g. approach to their translation. However, the categories of MWEs differ in their characteristics and impose different problems. The non-decomposable MWEs need to be defined in a dictionary in order to capture their proper translation. On the other hand, it is inefficient to add to the dictionary the decomposable MWEs as their number is large and their meaning is defined as a function of its constituents and based on that a translation approach can be designed in order to render the MWE in another language. Thus, in many cases we may be interested in discriminating between categories of MWEs.

The first method was found to give good results for the general task of MWEs recognition applied on the Brown Corpus of

Bulgarian – it gives 76% precision and 21% recall when we consider MWEs of frequency 5 or more and threshold of 0.3 for the MI value which is consistent with some results reported for other languages (Piao et al., 2005). It is also noted in the literature that statistical methods are not effective for identifying MWEs of low frequency and in fact about 50% of the MWEs occur 4 or less times in the corpus which is one of the main causes for the low recall value. However, by considering entities of 2 or more occurrences we improve slightly the recall (23%) but the precision is considerably reduced (64%).

The first method does not provide means to distinguish between categories of MWEs. In fact, the average MI values in the case of non-decomposable and decomposable MWEs are 0.31 and 0.35 respectively and the values are within similar ranges.

On the other hand the LSA method as applied to Brown Corpus of Bulgarian provides some relatively reliable measure for determining whether a MWE is compositional or not. In general, if similarity values measured between the MWE and each of its constituents are close to 1, this shows that the MWE is compositional and low similarity values show non-compositionality. The average measures of similarity between the constituents and the whole MWE or free phrase are presented in Table 1.

Similarity value with the MWE	Non-decomposable	Idiosyncratically decomposable	Decomposable	Free (non-MWE) phrase
Head word (N)	0.38	0.65	0.79	0.56
Constituent (A)	0.31	0.48	0.61	0.50

Table 1: Results from LSA: average similarity values between the MWE and its constituents for different types.

However, it is evident that the LSA is not sufficient to distinguish MWEs from non-MWEs as the similarity values for the free phrases fail to definitely reflect their full decomposability. Thus, in this case it is necessary to apply a preliminary method for determining MWEs and then LSA for their categorization.

It is also necessary to note that the ranges of similarity values for idiosyncratically decomposable and simple decomposable MWEs overlap considerably and this makes distinguishing between them difficult. For example, the named entity *Cherno more* (EN *Black sea*), has similarity value 0.96 for the head *more* (EN *sea*) and 0.81 for *cherno* (EN *black*). It is however not fully decomposable although the head of the phrase is a descriptor word which is reflected by the large similarity value but the adjective is not used in its general meaning.

3.2. Characteristics of corpora – size and domain

The type and size of corpora for the application of the methods also influences highly their performance. There are several factors that have particular importance: register and domain of corpora and their size.

The distribution of MWEs of different categories and free phrases varies across registers and domains. This can be observed from Table 2. The results are obtained on relatively small domain specific corpora (of about 20 000 words each) and the general Brown Corpus of Bulgarian.

domain, register / type	General	Administrative	Science	News	Informal
Non-decomposable MWEs	2.2%	0.6%	6.9%	4.2%	1.3%
Idiosyncratically decomposable MWEs	7.7%	6.7%	10.7%	6.4%	8.9%
Decomposable MWEs	33.2%	36.1%	39.9%	32.8%	11.0%
Free phrases	56.9%	56.6%	42.5%	56.6%	77.8%

Table 2: Distribution of categories MWEs and free phrases of the form adjective – noun across domains.

We need to consider these specific features when building resources for particular purposes because in some cases the occurrences of the event under observation may not be of sufficient number to be able to draw any conclusions. When using a general corpus for the purposes of automatic MWE recognition the proportion of texts from various domains needs to be taken into consideration.

The observations also show that the average frequency of the MWEs is almost twice as much as that of the free phrases. In the general corpus MWEs occur with average frequency of 5.1 and the free phrases with average frequency of 2.7. Domain specific corpora show similar results. In fact, there is a large number of free phrases which occur only once in the corpus and only a small number of MWEs with that frequency. We observe a tendency for repetition of MWEs which is expected in the case of scientific terms or other domain specific concepts and the opposite tendency for avoiding repetition in the case of free phrases ensuring diversity of the text. However, present observations are not sufficient to generalize these conclusions and are only limited to particular phrase structures.

It is also important to note that for particular domain specific entities the results obtained from a general corpus and a specialized corpus may differ significantly. For example, for the MWE *tanak klient* (EN *thin client*) using the first method we obtain mutual information measures of 0.96 and 0.42 from a special domain and a general corpus respectively. The same is evident from the LSA method application where we have high similarity values between the MWE and the head and the adjective for the special domain corpus (0.93 and 0.91) and significantly lower values from the general corpus (0.64 and 0.40).

3.3. Available annotation of analysed corpora

The first method requires POS tagging in order to facilitate the application of the syntactic filter. The LSA method requires lemmatization to be performed prior to analysis. However, for the application of LSA a word sense annotated corpus (a semantically annotated corpus is being developed for Bulgarian, see Koeva et al. 2006) can also be used and the vector space can be defined using not single words but senses corresponding to synsets in WordNet which will provide semantically substantiate measure for similarity.

Baldwin et al. (2003) also describe a possible approach to testing the results from the MWE identification using LSA by verifying the results with WordNet. They note that in most cases the decomposable MWEs are endocentric, i.e. a hyponym of their head word (Haspelmath 2002).

3.4. Specific features of Bulgarian

Some specific features of Bulgarian also influence highly the quality of the MWE recognition and need to be taken into account.

In the present study we only consider noun phrases of the form adjective – noun but in fact the MWEs in Bulgarian exhibit a very diverse structure and include examples of all syntactic classes: nouns, adjectives, adverbs, verbs. In this respect it is also important to consider the inflectional paradigms of MWEs (Koeva 2005) and in particular the cases where some restrictions apply.

Another challenge is to investigate the syntactic alternations which occur with some categories MWEs, especially simple decomposable and the cases of non-adjacent components of the entity where another phrase is possible to appear between components.

4. Conclusions and further work

Here we discussed some of the main factors influencing the performance of two methods for automatic MWE recognition and annotation. We need to emphasize that the results presented here are only valid for noun phrases of the form adjective – noun and the possible generalization of the observations over the whole group of MWEs is due to be evaluated.

However, we can conclude that the two approaches described can potentially be developed into a successful methodology by considering the parameters of the particular tasks – whether we need to simply identify MWEs or discriminate between categories. It is also important to consider the characteristics of the resources as they influence highly the results and take into account the specific features of the analysed corpora.

The extensive application and testing of methods for MWE identification remains one of the major tasks for Bulgarian.

5. Acknowledgements

The research presented here is developed with the financial support granted under Contract No. BG051PO001-3.3.04/27 of 28 August 2009 within the Operation Support to the development of PhD students, post-doctoral students, post-graduate students and young scientists of the General Directorate Structural Funds and International Educational Programmes with the Ministry of Education, Youth and Science, Bulgaria.

References

Baldwin, T., C. Bannard, T. Tanaka and D. Widdows *An Empirical Model of Multiword Expression Decomposability*. In: Proceedings of the ACL Workshop on Multiword Expressions: Analysis, Acquisition and Treatment, Japan, 2003. http://lingo.stanford.edu/pubs/tbaldwin/acl2003mwe-decomposability.pdf

Deerwester, S., S. Dumais, G. Furnas, T. Landauer and R. Harshman *Indexing by Latent Semantic Analysis.* In: Journal of the Society for Information Science, 41(6), 1990, pp. 391-407.

Haspelmath, M. *Understanding morphology.* Arnold publishers, London, 2002.

Justeson, J. and S. Katz *Technical terminology: some linguistic properties and an algorithm for identification in text.* // Natural Language Engineering, 1995, pp. 9-27.

Manning, C. and H. Schutze *Foundations of Statistical NLP*. MIT Press, 1999.

Katz, G. and E. Giesbrecht *Automatic identification of non-compositional multi-word expressions using latent semantic analysis.* In: Proceedings of the workshop on MWEs: Identifying and exploiting underlying properties, Sydney, 2006, pp. 12-19.

Koeva, S. *Inflection Morphology of Bulgarian Multiword Expressions.* In: Computer Applications in Slavic Studies – Proceedings of Azbuki@net, International Conference and Workshop, Sofia, 2005, pp. 201-216.

Koeva, S. *Multi-word term extraction for Bulgarian.* In: Balto-Slavonic Natural language Processing, Prague, 2007, pp. 59-66.

Koeva, S., S. Leseva, I. Stoyanova, E. Tarpomanova and M. Todorova. *Bulgarian Tagged Corpora*, Proceedings of the Fifth International Conference Formal Approaches to South Slavic and Balkan Languages, 18-20 October 2006, Sofia, Bulgaria, pp. 78-86.

Piao, S., P. Rayson, D. Archer, T. McEnery *Comparing and combining a semantic tagger and a statistical tool for MWE extraction.* Computer Speech and Language, (Special issue on Multiword expressions), Volume 19, 2005, issue 4, pp. 378 - 397, Elsevier.

A COMPUTATIONAL MODEL OF CROATIAN DERIVATIONAL MORPHOLOGY

Jan Šnajder and Bojana Dalbelo Bašić

University of Zagreb, Faculty of Electrical Engineering and Computing
Unska 3, Zagreb, Croatia
{jan.snajder, bojana.dalbelo}@fer.hr

ABSTRACT

Computational models of derivational morphology of Slavic languages have attracted much interest in recent years. This paper describes a computational model of Croatian derivational morphology, limited at present to suffixal derivation. The model extends on the previously developed model of Croatian inflectional morphology, and uses higher-order functions to succinctly define the word formation rules. The basic building blocks of the model are derivational patterns, closely resembling the derivational patterns found in traditional grammar books. We describe how the model can be used to create derivational families of Croatian words, and discuss the problem of spurious derivations.

1. Introduction

Computational models of morphology have a long and established tradition within computational linguistics (Sproat 1992). Such models find various applications in task related to natural language processing, such as information retrieval, text analytics, ontology construction, machine translation, etc. Morphology of Slavic languages, being complex regarding both inflection and derivation, poses a real challenge for computational modeling. At the derivational level, the morphological complexity manifests itself in a large number of derivational patterns, reflecting the many ways in which new words can be derived from existing words.

Much recent work has focused on computational models of derivational morphology of Slavic languages and the analysis of morpho-semantic properties of derivational relations. As noted by Pala (2008), the semantic nature of derivational relations needs a systematic examination before these relations can be fully exploited in NLP applications, and there has in fact been much recent work in this direction (Koeva et al. 2008; Azarova 2008; Pala et al. 2007). Notably, a framework has been suggested for a more systematic treatment of derivational relations of Slavic languages (Pala 2008). A necessary prerequisite, however, is a computational tool capable of processing derivational relations for a given language. The work described in this paper aims at developing such a tool for Croatian language: we describe a computational model of Croatian derivational morphology, restricted at present to suffixal derivation. The model builds and extends on a previously developed word-based morphology model of inflectional morphology described by Šnajder and Dalbelo Bašić (2008). Unlike the previously developed computational models of Croatian morphology (Lopina 1992; Tadić 1994; Ćavar et al. 2008), our model focuses on derivational morphology.

The paper is structured as follows. In the next section we define formally the derivational model of morphology and describe briefly its implementation. Section 3 describes the Croatian derivational patterns implemented by the model. In Section 4 we describe how the model can be used to generate the so-called derivational families of Croatian words and discuss the problem of spurious derivations. Section 5 concludes the paper and outlines future work.

2. Model of derivational morphology

To make morphology modeling less tedious, our primarily aim was to devise a formalism that closely resembles the grammar descriptions as found in traditional grammar books. The basic building blocks of our model of derivational morphology are the *derivational patterns* and *word formation rules*. We first briefly describe the underlying model of inflectional morphology.

2.1. Underlying inflectional model

The model of derivational morphology extends the inflectional model described in (Šnajder and Dalbelo Bašić 2008). The inflectional model defines a set of inflectional patterns, each of which defines how a word's stem can be transformed into the corresponding word forms. An important feature of this model is that is both generative and reductive – it can be used to generate the word forms of a given stem as well as to reduce a word form back to its stem. What this means in practice is that the lemmas can be feed as the inputs to the model, rather than the stems. In order to generate word forms from a lemma, the inflectional model works in a reduce-than-generate fashion: given a lemma and an inflectional pattern, the model first reduces the lemma to the corresponding stem, and then generates the word forms from the so-obtained stem. Thus what the model expects as the input are the lemma-pattern pairs (LP-pairs for short). A lemma-based model is arguably more convenient than a stem-based model because lemmas are more easily readable, but also because, as we show below, a lemma-based model simplifies the modeling of derivation.

2.2. Derivational patterns

A derivational pattern describes how a new word of a different semantic category can be derived from an original word. A derivational pattern typically defines: (1) the word formation rule, which defines how the original word's (derivational) stem is transformed to obtain the new word, (2) the lexical (sub)category of the original word, and (3) the lexical and semantic category of the derived word. The derivational stem is usually the same as the inflectional stem. The semantic category of the derived word depends on its lexical category; e.g., the semantic category of a noun may be either *agent*, *demonym*, *location*, etc.

Within our model, each grammatical category is represented by a set of corresponding inflectional patterns. For example, the lexical category *possessive adjective* is represented by a set of all inflectional patterns that define the inflection of possessive adjectives. Such sets defining the lexical categories are associated with both the original and the derived word. The semantic category of the derived word is not explicitly modeled in the current version of the model (though this extension should be rather straightforward).

Let $\mathcal{F}$ be a set of inflectional patterns defined by the inflectional model, let $\mathcal{S}$ be a set of stems and word forms, and let $\mathcal{T}$ be a set of word formation rules, which we call the *transformation functions* (described in the next subsection). A derivational pattern d is a triple:

$$d = (t, \mathcal{F}_1, \mathcal{F}_2) \ \in \ \mathcal{T} \times \wp(\mathcal{F}) \times \wp(\mathcal{F})\,. \tag{1}$$

Function t is the transformation function by which the inflectional stem of the original word is transformed into the lemma of the derived word. Sets of inflectional rules $\mathcal{F}_1$ and $\mathcal{F}_2$, where $\mathcal{F}_1 \subseteq \mathcal{F}$ and $\mathcal{F}_2 \subseteq \mathcal{F}$, define the lexical category of the original and derived word, respectively. Let $\mathcal{D}$ be the set of the derivational patterns.

Note that the word formation rule, given by transformation function t, is asymmetric in the sense that it operates on the original word's stem to produce a derived word's lemma. Thus, for example, to derive *prijatelj* $\rightarrow$ *prijateljica*, we define $t = sfx(ica)$. A more linguistically plausible approach – one that more clearly separates inflection from derivation – would be to have a stem-to-stem formation rule t, which operates on a stem and also derives a stem, and then subsequently applies the inflectional model to generate the lemma (or any other word form). In this case we would have $t = sfx(ic)$ instead of $t = sfx(ica)$, because *-a* is an inflectional morpheme. We chose to use the stem-to-lemma rules because (1) this is exactly how word formation rules are defined in the traditional grammar books , cf. (Babić 2002; Barić et al. 2005), and (2) stem-to-lemma rules reduce the ambiguity of derivation. The latter is the case because the lemma is more indicative of the correct inflectional pattern than a stem; a stem can easily be combined with wrong inflectional patterns and yield spurious word forms. Thus, using stem-to-lemma rules makes the model more comprehensive and more precise. For completeness let us note that the lemma-to-stem and lemma-to-lemma rules are also possible (due to the inflectional model being reductive), but these are clearly not convenient.

Also note that the transformation function t is defined with respect to the inflectional rather than derivational stem. In cases when these two differ (as is the case with some verb-motivated patterns in Croatian language) the transformation itself should compensate for the difference (by enclosing a transformation that transforms the inflectional stem to the derivational stem).

2.3. Word formation rules

At the word formation level, our model uses functions to represent the individual word formation rules. An important practical issue is how to conveniently define such functions in a computational model. The representation formalism – much as any other formalism – should strike a balance between expressivity and simplicity: it should be able to represent most, if not all, word formation rules of a given language, but at the same time the formalism should correspond closely to the simple, human-readable descriptions found in traditional grammar books. On top of that, the formalism should be computationally feasible. To suit these requirements, we use the Higher-Order Functional Morphology (HOFM) representation formalism, introduced by Šnajder and Dalbelo Bašić (2008) and extended by Šnajder (2010).

The HOFM formalism makes use of higher-order functions – a notion drawn from functional programming languages (Hudak 1989) – to represent the individual word formation rules. A word formation rule is represented by a *transformation function* $t : \mathcal{S} \to \wp(\mathcal{S})$, mapping from a set of strings $\mathcal{S}$ (including word forms, stems, and affixes) to a set of transformed strings (unless the transformation is ambiguous, the resulting set contains a single string). A transformation t may not be applicable to a given string s, which is indicated by $t(s) = \emptyset$. A higher-order function of the form $f : X \to \mathcal{T}$ is used to define a transformation function parametrized by a value from X. For example, a higher-order function $sfx : \mathcal{S} \to \mathcal{T}$ may be used to define suffixation transformation, with $sfx(s)$ yielding a transformation function of suffixing the string s to a word's stem. If we define, for example, $t = sfx(ica)$, we can use t in a derivational pattern as a word formation rule that suffixes *-ica* to a stem, e.g., $t(prijatelj) = \{prijateljica\}$. The basic idea is to use one higher-order function for each distinct type of morphological transformation (suffixation, prefixation, phonological alternations). More complex word formation rules, such as those combining prefixation and suffixation, can then be obtained straightforwardly by functional composition. For example, if we define $t = sfx(ica) \circ pfx(ne)$, then $t(prijatelj) = \{neprijateljica\}$. Note that transformation functions operate directly on surface-forms (unlike the two-level morphology models), thus the order of composition matters.

One important feature of the HOFM formalism is that transformations can capture nonfunctional relations, allowing us to represent ambiguous word formation rules. Ambiguous transformations may be used to model grammar ambiguities, i.e., the cases where one derivational pattern admits more than one derivation. We can think of an ambiguous transformation as a transformation that can choose among two or more (possibly also ambiguous) transformations. To define an ambiguous transformation, we use an *(impartial) choice operator*, $| : \mathcal{T} \times \mathcal{T} \to \mathcal{T}$. E.g., an ambiguous transformation that can choose between transformations t_1 and t_2 is defined as $t_1 \,|\, t_2$. As a shorthand, we define an optionality operator, $opt : \mathcal{T} \to \mathcal{T}$, as $opt(t) = t \,|\, nul$, where nul is the identity transformation. Besides for modeling of ambiguity, the choice operator may also used for modeling of phonological alternations. Phonological alternations may be represented as a sequence of (mutually exclusive) choices among individual suffix alternations. For example, sibilarization may be defined as $plt = rsfx(k, c)|rsfx(g, z)|rsfx(h, s)$, where $rsfx(s_1, s_2)$ is a higher-order function that replaces string suffix s_1 with string suffix s_2 (the word *suffix* here is not used in the strict linguistic sense).

Another type of choice operator that is used in HOFM is the *biased choice operator*, $|| : \mathcal{T} \times \mathcal{T} \to \mathcal{T}$. If a transformations is defined as $t_1 \,||\, t_2$, it will first attempt to transform a string using t_1. Only if this attempt fails, will the transformation t_2 be applied. As a shorthand, we define the *attempt operator* as $try(t) = t \,||\, nul$. We can think of transformation $try(t)$ as conditional in the sense that it will be applied only if it can be applied to a given string, otherwise the string will remain unaltered.

2.4. Derivation

We next describe how a derivational pattern $d \in \mathcal{D}$ is used to generate derivations of words. Recall that the underlying inflectional model is lemma-based and uses LP-pairs as input. A compatible model of derivational morphology must therefore take LP-pairs as input. When generating the LP-pairs, we must account for ambiguity in the word formation rules. Thus, given a single derivational rule and a single LP-pair, our model will derive a set of possible derivations instead of a single derivation.

Let $(l, f) \in \mathcal{S} \times \mathcal{F}$ be an LP-pair. The derivation of LP-pairs is formalized by the function $lDerive : \mathcal{S} \times \mathcal{F} \times \mathcal{D} \to \wp(\mathcal{S} \times \wp(\mathcal{F}))$, defined as follows:

$$lDerive\big(l_1, f_1, (t, \mathcal{F}_1, \mathcal{F}_2)\big) = \begin{cases} \big\{(l_2, \mathcal{F}_2^*) \ : \ l_2 \in (t \circ t_0^{-1})(l_1)\big\} & \text{if } f_1 \in \mathcal{F}_1 \ \wedge \ \mathcal{F}_2^* \neq \emptyset\,, \\ \emptyset & \text{otherwise}\,, \end{cases} \tag{2}$$

$$\mathcal{F}_2^* = \{f_2 \in \mathcal{F}_2 \ : \ f_2 \vDash l_2\}\,. \tag{3}$$

Function t_0, which is defined by the inflectional pattern f_1 (abstracted here), defines the transformation of the original word's stem into the lemma l_1. Its inverse function, t_0^{-1}, defines the reduction of the original word's lemma into the inflectional stem (note that the inverse transformation is possible because the inflectional model is reductive). The word formation is computationally performed in two steps: first, the original word's lemma l_1 is reduced to the stem, and then the stem-to-lemma transformation t is applied to derive the lemma l_2. Because t_0 and t may be ambiguous, there may be more than one result for l_2. Next, each lemma l_2 is paired with the set of inflectional patterns $\mathcal{F}_2^*$, which define the lexical category of the derived word. Set $\mathcal{F}_2^*$ is obtained by choosing from set $\mathcal{F}_2$ the inflectional patterns that are applicable to lemma l_2. The applicability of inflectional pattern f_2 to lemma l_2 is denoted by '$f_2 \vDash l_2$'; for details the reader is referred to (Šnajder and Dalbelo Bašić 2008). If the inflectional pattern f_1 of the original word does not match the lexical category of the derivational pattern (i.e., if $f_1 \notin \mathcal{F}_1$), or if the transformation function is not applicable to the given lemma (i.e., if $(t \circ t_0^{-1})(l_1) = \emptyset$), the derivation fails and $lDerive$ evaluates to an empty set.

2.5. Derivational relation

Thus far we have focused on the generative aspect of our model and showed how the model can be used to derive new words. If we wish to analyze the derivational relations between existing words, we need to be able to check whether a derivational relation holds between a given pair of words. Relation testing can be accomplished with our generative model simply by attempting to derive one word from the other. Because our model is lemma-based and uses LP-pairs to represent the words, the test amounts to checking whether one LP-pair can be derived from the other.

We formalize this by introducing a (direct) derivational relation, denoted by '$\to_d$'. This is a binary relation on $\mathcal{S} \times \mathcal{F}$, parametrized by the derivational pattern d, and defined as follows:

$$(l_1, f_1) \to_d (l_2, f_2) \quad \iff \quad (l_2, \mathcal{F}_2^*) \in lDerive(d, l_1, f_1) \ \wedge \ f_2 \in \mathcal{F}_2^*\,. \tag{4}$$

It is important to understand that relation $\to_d$ does not imply actual derivational relatedness. The relation merely indicates that two LP-pairs are potentially derivationally related, in that (1) there exists a surface-form relation and (2) the lexical categories match. For lemmas to be derivationally related, the words must also be semantically related. It is the absence of semantic relation that our model is unable to capture. We go back to this issue in Section 4.2., when we discuss the problem of spurious derivations.

2.6. Implementation

Because our model is function-based, it is perhaps most easily implemented in a functional programming language. For this purpose, we have chosen the Haskell programming language (Jones 2003). Haskell, being a high-level and a typed language, is also a good choice for hosting a domain specific language. What this means is that in Haskell one can easily define functions, operators, and (to a limited extent) syntax so that embedded within Haskell we have a language focused on a specific problem domain. In our case, the domain specific language is used to define the model of derivational morphology, i.e., the derivational patterns and the transformation functions. This way the model is easier to create and understand.

For example, a transformation function $t = sfx(ak) \circ try(plt)$, which attempts to palatalize the stem and then adds to it the suffix *-ak*, is simply defined as follows:

```
t = sfx "ak" & try plt
```

were `&` is the composition operator and `sfx`, `try`, and `plt` are higher-order functions. We can now use `t` to define a pattern for the derivation of diminutive masculine nouns from masculine nouns, as follows:

```
d = DPattern "iu02" t mNouns mNouns
```

where `mNouns` is a set of inflectional patterns for masculine nouns, and string `iu02` is the label of the pattern. We now can feed pattern `d` as the input to the `lDerive` function. For example:

```
> lDerive d ("smijeh",n04)
  [("smiješak",[N28,N47])]
> lDerive d ("cvijet",n04)
  [("cvijetak",[N28,N47])]
```

where `N28` and `N47` are the inflectional patterns defined by the inflectional model. Note that palatalization was not applied in the second case because it is not applicable to the given stem.

3. Croatian derivational patterns

The model of Croatian derivational morphology consists of 244 patterns, which describe most of suffixal derivation of Croatian nouns, verbs, and adjectives. The pattern definitions were, with minor modifications, adopted from (Barić et al. 2005). We have grouped the derivational patterns are according to the semantic category of the derived word (with some simplifications). Table 1 shows a breakdown of the derivational patterns according to the semantic categories. Note that the current version of the model is restricted to suffixal derivation, which for nouns and adjectives happens to be the most productive (Barić et al. 2005).

3.1. Modeling phonological alternations

Phonological alternations are frequent in Croatian derivational morphology, modeling of which presents a challenge in its own right. Phonological alternations – both at the inflectional and the derivational level – may be either phonologically or morphologically conditioned. The former are applied universally, whereas the latter are applied depending on the morphological category of the word form.

Phonologically conditioned alternations are built in the model in such a way that their application is attempted (using the operator try) before any application of a suffixation transformation. Thus phonologically conditioned alternations need not be modeled explicitly in the derivational patterns. For example, the pattern for deriving possessive adjective *klub* $\rightarrow$ *klupski* is defined simply as:

Table 1: Groups of Croatian derivational patterns included in the model

Group	Semantic category of the derived word	Num. of patterns	Example
N-1	Masculine agent nouns	21	*banka* → *bankar*
N-2	Masculine noun expressing a characteristic	5	*sretan* → *sretnik*
N-3	Masculine nouns for a follower	3	*Franjo* → *franjevac*
N-4	Female person nouns	11	*prijatelj* → *prijateljica*
N-5	Nouns for male and female person	5	*izdati* → *izdajica*
N-6	Demonyms and ethnonyms	11	*Varaždin* → *Varaždinac*
N-7	Nouns for animals and plants	6	*otrovan* → *otrovnica*
N-8	Nouns for inanimate objects derived from nouns and verbs	11	*mijenjati* → *mjenjač*
N-9	Nouns for places	9	*cigla* → *ciglana*
N-10	Abstract nouns	18	*prijatelj* → *prijateljstvo*
N-11	Deverbal (action) nouns	24	*čuvati* → *čuvanje*
N-12	Diminutives and augmentatives	19	*orah* → *oraščić*
N-13	Collective nouns	7	*radnik* → *radništvo*
N-14	Other types of nouns	6	*brod* → *brodarina*
A-1	Qualifying adjectives	35	*mrak* → *mračan*
A-2	Possessive adjectives	19	*djed* → *djedov*
A-3	Passive verb adjectives	9	*spasiti* → *spašen*
V-1	Imperfective verbs	12	*baciti* → *bacati*
V-2	Diminutive and pejorative verbs	6	*govoriti* → *govorkati*
V-3	Verbs derived from nouns	5	*večera* → *večerati*
V-4	Verbs derived from adjectives	2	*sitan* → *sitniti*

```
DPattern "ppo6" (sfx "ski") nouns pAdjectives
```

because the alternation *b/p* will be applied implicitly. It should be noted that phonologically conditioned alternations occur more frequently with prefixal derivation, which is not yet included in the model.

Unlike the phonologically conditioned alternations, morphologically conditioned alternations are modeled explicitly as conditional transformations using the operator try described earlier. This is because one and the same derivational pattern must also be applicable to stems for which the specific phonological alternation is not applicable. For example, the derivation *onečistiti* → *onečišćavati* is achieved by jotating *st/šć* and by suffixing *-avati*, whereas in *odobriti* → *odobravati* jotation is not applicable. The derivational pattern is therefore defined as:

```
DPattern "gv03" (sfx "avati" & try jot) tiVerbs tiVerbs
```

3.2. Modeling ambiguities

Two types of ambiguities are encountered when modeling word formation rules of Croatian language. First type is the ambiguity of a word formation rule: given a word (a stem), two or more equally valid derivations are possible. Such cases arise most often with the so-called reflex of jat alternation (*ije/je*, *ije/e*, and *ije/i*). As an example, consider *brijeg* → *bregovit/brjegovit*, for which the corresponding derivational pattern is defined as:

```
DPattern "po21" (sfx "ovit" & try (rifx "ije" "je" .|. rifx "ije" "je")) mNouns qAdjectives
```

where `rifx` is a higher-order function for substring replacement and `.|.` is the the choice operator.

Another type of ambiguity arises from grammar inconsistencies. For example, when deriving agent nouns from nouns by suffixing *-ar*, besides alternation *ije/je* (e.g., *mlijeko* → *mljekar*), the stem is jotated in some

cases (e.g., *tvornica* → *tvorničar*), whereas it is unaltered in others (e.g., *biblioteka* → *bibliotekar*). The corresponding derivational pattern is thus defined as:

```
DPattern "iv16" (sfx "ar" & try (rifx "ije" "je") & opt jot) (fNouns++nNouns) mNouns
```

4. Using the model

4.1. Building the derivational families

Given an adequate lexicon (i.e., a list of LP-pairs) the model of derivational morphology can be used to group together the (potentially) derivationally related words, forming (potential) derivational families or nests. These are *potential* because, as noted in section 2.5., the derivational relation necessitates the semantic relatedness, the absence of which cannot be detected by our model. Formally, the derivational families correspond to equivalence classes of the derivational relation given by (4). That is, if $\mathcal{L} = \mathcal{S} \times \mathcal{F}$ is a lexicon (a set of LP-pairs), the set of derivational families is the quotient set $\mathcal{L}/{=_{\mathcal{D}}}$, where $=_{\mathcal{D}}$ is the reflexive and symmetric closure of relation $\overset{*}{\rightarrow}_{\mathcal{D}}$ that is defined as follows:

$$w_1 \overset{*}{\rightarrow}_{\mathcal{D}} w_2 \quad \Longleftrightarrow \quad \exists d \in \mathcal{D}.\left((w_1 \rightarrow_d w_2) \ \vee \ \exists w_3 \in \mathcal{L}.\left((w_1 \rightarrow_d w_3) \ \wedge \ (w_3 \overset{*}{\rightarrow}_{\mathcal{D}} w_2)\right)\right). \tag{5}$$

It should be noted that the transitivity is ensured over existing LP-pairs from the lexicon. As a consequence, poor lexicon coverage may result in fragmented derivational families.

The quotient set $\mathcal{L}/{=_{\mathcal{D}}}$ may be computed efficiently by computing the weakly connected components of a strongly connected digraph. In this graph, the vertexes correspond to LP-pairs from the lexical database and unidirectional arcs connect the original and the derived LP-pairs. By computing the quotient set in this manner, the direction of derivation need not be inverted, which would otherwise be required for computing the symmetric closure of $\overset{*}{\rightarrow}_{\mathcal{D}}$. (Note, however, that inverting the application of $lDerive$ function is not really a problem because HOFM transformation functions are invertible).

As a proof of concept, we have constructed potential derivational families from a lexicon consisting of 47,415 Croatian words. The lexicon was acquired automatically from an unannotated newspaper corpus using the inflectional morphology model; for details the reader is referred to (Šnajder et al. 2008). From this lexicon we have obtained 34,310 derivational families. The average size of a potential derivational family was 1.38, and the maximum size was 53. Actually, there were two families of size 53, both over-inflated due to spurious derivations (see below). Smaller families (10 words or less), and especially those with longer stems, seem to correspond well to true derivational families.

4.2. Spurious derivations

A limitation of our model is that it may generate *spurious derivations* – derivations in which the original word and the derived word are not semantically related. There are two possible reasons why a surface-form valid derivation may lack semantic relatedness. The first are the homographic stems, i.e., stems sharing identical forms but having different meanings. Examples are the spurious derivations *šal* → **šalica* (*scarf* → **mug*), *nos* → **nositi* (*nose* → **to carry*), and *vod* → **voda* (*squad* → **water*); the corresponding examples of valid derivations are *prijatelj* → *prijateljica*, *rod* → *roditi*, *kum* → *kuma*, respectively. Another, less obvious reason why derivation may be spurious is when the original and the derived word are etymologically but not semantically related, as is the case with *nov* → *novac* (*new* → **money*) and *stol* → *stolac* (*table* → *chair*). For a more thorough discussion of this issue the reader is referred to (Babić 2002).

To gain some preliminary insight into the extent of spurious derivations, we have conducted an experiment as follows. The aim was to investigate which derivational patterns are most prone to spurious derivations. To this end, we have built a sample of 3,307 derivational families from the Culture section of the Croatian newspaper

Table 2: Analysis of the ten most frequently applied derivational patterns

			Derivations		
Group	Suffix	Example	Valid	Spurious	(%)
A-1	*-an*	*beskraj* → *beskrajan*	108	9	7.7
A-3	*-en*	*dogovoriti* → *dogovoren*	79	0	0
N-11	*-nje*	*čitati* → *čitanje*	64	0	0
A-2	*-ski*	*autor* → *autorski*	43	1	2.3
N-10	*-ost*	*aktivan* → *aktivnost*	40	2	4.8
A-3	*-an*	*čitati* → *čitan*	37	3	7.5
N-10	*-ost*	*duhovit* → *duhovitost*	28	0	0
N-11	*-enje*	*donositi* → *donošenje*	23	0	0
N-11	*-a*	*nagraditi* → *nagrada*	21	1	4.6
V-1	*-ati*	*javiti* → *javljati*	18	2	10.0

Vjesnik. The derivational families were hand-validated using (Anić 2003) as an authoritative reference source. The average number of LP-pairs per family was 1.46, and the maximum number was 11 (derivational family motivated by the word *stvar*). Table 2 shows the analysis for the ten most frequently applied derivational patterns. Results suggest that some derivational patterns are more prone to spurious derivations than others. For most patterns, however, spurious derivations are rather rare.

It should be noted that, although spurious derivations are relatively rare, automatically constructed derivational families easily over-inflate because they are constructed by transitive closure. Therefore, if one wishes to attain perfect quality (e.g., for lexicography), potential derivational families need to be split up manually. If perfect quality is not mandatory (e.g., for information retrieval), one can attempt to minimize spurious derivations by constraining in various ways the application of certain derivational patterns. We leave this as a subject for future investigation.

5. Conclusion

We have described a computational model of Croatian derivational morphology. The basic building blocks of the model are derivational patterns, closely resembling the derivational patterns found in traditional grammar books. The model builds on a previously developed inflectional model, and uses higher-order functions to succinctly define the word formation rules. The model at present is limited to suffixal derivation, but it can be easily extended to cover prefixal derivation as well. We have illustrated how the model can be used to generate potential derivational families of Croatian words. A preliminary investigation of the extent of spurious derivation suggests that they occur rather rarely.

The model may be used in a wide range of applications, ranging from language study and (computational) lexicography to natural language processing and information retrieval. In particular, we hope the described derivational model may prove itself useful in a more systematic examination of the semantic properties of Croatian derivational relations, preferably within a wider framework such as the one proposed by Pala (2008).

For future work, we plan to extend our model with prefixal derivation. An interesting line of research would be to combine the model of derivational morphology with a computational model of semantics, such as the distributional semantic model (Lenci 2008). This could provide us with interesting insights into the semantics of derivational relations.

Acknowledgments

This work has been supported by the Ministry of Science, Education and Sports, Republic of Croatia under the Grant 036-1300646-1986. The authors thank the anonymous reviewer for his or her useful comments.

References

Anić, V. 2003. *Veliki rječnik hrvatskoga jezika*. Novi Liber.

Azarova, I. 2008. Derivational semantic relations in RussNet. In *Proceedings of the 4th Global Wordnet Conference*, Szegéd.

Babić, S. 2002. *Tvorba riječi u hrvatskome književnome jeziku*. HAZU, 3rd ed.

Barić E.; Lončarić M.;, Malić, D.; Pavešić, S.; Peti, M.; Zečević, V.; Znika M. 2005. *Hrvatska gramatika*. Školska knjiga, 4th ed.

Ćavar D.; Jazbec, I-P.; Stojanov T. 2008. CroMo – morphological analysis for standard Croatian and its synchronic and diachronic dialects and variants. In *Proceedings of FSMNLP 2008*, Ispra, Italy.

Hudak, P. 1989. Conception, evolution, and application of functional programming languages. *ACM Computing Surveys*, 21(3):359–411.

Jones, S. P. 2003. Haskell 98 language and libraries: The revised report.

Koeva S.; Krsteva C.; Vitas D. 2008. Morpho-semantic relations in Wordnet – a case study for two Slavic languages. In *Proceedings of the 4th Global Wordnet Conference*, pages 239–253, Szegéd.

Lenci A. 2008. Distributional semantics in linguistic and cognitive research. *Italian Journal of Linguistics*, 20 (1):1–31.

Lopina V. 1992. Dvorazinski opis morfonoloških smjena u pisanome hrvatskom jeziku. *Suvremena lingvistika*, 34:185–194.

Pala K. & Hlaváčková D. 2007. Derivational relations in Czech WordNet. In *Proceedings of the Workshop on Balto-Slavonic Natural Language Processing, ACL*, pages 75–81, Prague.

Pala K. 2008. Derivational relations in Slavonic languages. In *Proceedings of the 6th International Conference on Formal Approaches to South Slavic and Balkan Languages, FASSBL6*, pages 21–28, Dubrovnik, Croatia.

Sproat R. 1992. *Morphology and Computation*. MIT Press.

Šnajder J. 2010. *Morfološka normalizacija tekstova na hrvatskome jeziku za dubinsku analizu i pretraživanje informacija*. PhD thesis, University of Zagreb, Faculty of Electrical Engineering and Computing, Zagreb.

Šnajder J. & Dalbelo Bašić B. 2008. Higher-order functional representation of Croatian inflectional morphology. In *Proceedings of the 6th International Conference on Formal Approaches to South Slavic and Balkan Languages, FASSBL6*, pages 121–130, Dubrovnik, Croatia.

Šnajder J.; Dalbelo Bašić B.; Tadić M. 2008. Automatic acquisition of inflectional lexica for morphological normalisation. *Information Processing and Management*, 44(5):1720–1731.

Tadić M. 1994. *Računalna obrada morfologije hrvatskoga književnoga jezika*. PhD thesis, University of Zagreb, Faculty of Humanities and Social Sciences, Zagreb.

VERB VALENCY FRAME EXTRACTION USING MORPHOLOGICAL AND SYNTACTIC FEATURES OF CROATIAN

Krešimir Šojat*, Željko Agić, Marko Tadić***

*Department of Linguistics, **Department of Information Sciences
Faculty of Humanities and Social Sciences, University of Zagreb
Ivana Lučića 3, 10000 Zagreb, Croatia
{ksojat, zeljko.agic, marko.tadic}@ffzg.hr

ABSTRACT

The paper presents an approach to valency frame extraction for Croatian verbs on basis of morphological and syntactic features of wordforms from syntactically annotated sentences. We have used a gold standard sample of approximately 1200 sentences and 30.000 tokens from the Croatian Dependency Treebank and a frame instance extraction algorithm. We extracted 936 verb frame instances for 424 different verbs – consisting of lemmas, morphosyntactic tags and syntactic functions of the encountered wordforms – and manually assigned tectogrammatical functors to their elements. Distributional properties are given in terms of co-occurrences for each of these features. The obtained results will serve for further development of valency frame extraction procedures.

1. Introduction

Recent enhancements of the Croatian Dependency Treebank both in size and annotation quality enabled the development of procedures for (semi-)automatic extraction of valency frames for Croatian verbs. The initial experiment, presented in (Agić et al. 2010), produced a rule-based procedure for the extraction of specific instances of verb valency frames from the treebank. On the basis of the results shown there, we present in this paper an extension of that specific line of research in terms of improvements of the algorithm to be used in further semi-automatic construction of verb valency frames. More specifically, in this paper we attempt to induce a set of statistically verified rules for the assignment of the most probable tectogrammatical functors to sentence elements on the basis of their morphosyntactic features and syntactic functions. In order to achieve the objective we extracted valency frame instances for verbs from a gold standard section of the treebank, manually annotated the extracted elements for tectogrammatical functors and established a set of relations between verbs, tectogrammatical functors (that roughly correspond to the notion of semantic or theta roles), syntactic functions and morphosyntactic features in the form of statistical distributions of their co-occurrences. We hope to use these distributional properties in the process of semi-automatic valency frame induction by applying the acquired rules on unseen portions of the treebank. To the best of our knowledge, other than (Agić et al. 2010) and (Šojat et al. 2010) – the latter implementing a rule-based approach to valency – no similar experiments in verb valency frame extraction were done on Croatian texts.

In the following section of the paper, we present the recent advancements in the development of Croatian Dependency Treebank in more detail. Sections 3 and 4 present the setup of the experiments and discuss the obtained results. We conclude the paper with an outline of future research, specifically emphasizing the utilization of the treebank and verb valency lexicons in stochastic dependency parsing.

2. Croatian Dependency Treebank

Croatian Dependency Treebank (hr. *Hrvatska ovisnosna banka stabala*, HOBS further in the text), as described in e.g. (Tadić 2007) and (Agić et al. 2010), is a dependency treebank built along the principles of Functional Generative Description (FGD) (Sgall et al. 1986), a multistratal model of dependency grammar developed for Czech. In a somewhat simplified version, the FGD formalism was further adapted in the Prague Dependency Treebank (PDT) (Hajič et al. 2000) project and applied for the sentence analysis and annotation on the levels of morphology, syntax – in the form of dependency trees with nodes labeled with syntactic functions – and tectogrammatics.

Annotation of a sentence at the morphological layer consists of attaching several attributes to the tokens such as morphological lemmas and morphosyntactic tags. At the analytical layer, the sentence is represented in the form of a tree with labeled nodes. In the syntactic analysis of a sentence a set of analytical functions such as subject or object are attached

to nodes of the tree as attributes. On the tectogrammatical layer, i.e. on the layer of the representation of sentence meaning and semantic relations among its elements sentences are also represented as rooted trees with labeled nodes. Unlike the analytical layer, not all the morphological tokens are represented at the tectogrammatical layer (e.g. there are no prepositions, nodes representing omitted subject are introduced, etc.). Similarly to the analytical layer, the edges of the tree represent relations between the nodes, the type of the relation being indicated by a set of labels. The total of 39 attributes can be assigned to every non-root node of the tectogrammatical tree. Every node representing a verb or a certain type of a noun has a valency frame assigned to it by means of a reference to a valency dictionary PDT-VALLEX (Hajič et al. 2003) (cf. http://ufal.mff.cuni.cz/pdt2.0/).

The ongoing construction of HOBS closely follows the guidelines set by the PDT, with their simultaneous adaptation to the specifics of the Croatian language. More detailed account of the HOBS project plan is given in (Tadić 2007). HOBS at this moment (2010-09) consists of approximately 2.870 sentences in the form of dependency trees that were manually annotated with syntactic functions using TrEd (Pajas 2000) as the annotation tool, whereas the manual annotation of sentences on the tectogrammatical layer is currently not conducted. These sentences, encompassing approximately 70.000 tokens, stem from the magazine Croatia Weekly, i.e. the Croatia Weekly 100 kw (CW100) corpus that is a part of the newspaper sub-corpus of the Croatian National Corpus (HNK) (Tadić 2000). The Croatia Weekly sub-corpus was previously XCES-encoded, sentence-delimited, tokenized, lemmatized and MSD-annotated by linguists using a semiautomatic procedure (cf. Tadić 2002). Thus, each of the analyzed sentences contains the manually checked information on part-of-speech, morphosyntactic category, lemma, dependency and analytical function for each of the wordforms. Such a course of action, i.e. the selection of the corpus, was taken in order to enable the training procedures of various state-of-the-art dependency parsers (Buchholz et al. 2006), (Nivre et al. 2007), to choose from a wide selection of different features in this and the upcoming experiments with stochastic dependency parsing of Croatian texts. Basic stats for HOBS and the experiment sets are given in Table 1 and will be further discussed in the following section. Sentences in HOBS are annotated according to the PDT annotation manual for the analytical level of annotation, with respect to differing properties of the Croatian language and consulting the Slovene Dependency Treebank (SDT) project (Džeroski et al. 2006). The utilized analytical functions are thus compatible with those of the Prague Dependency Treebank. Further work on HOBS includes, among other tasks: enlarging the treebank, cross-validating the treebank annotation, designing a manual for HOBS annotation and conducting a comparative analysis of HOBS, SDT and PDT.

3. Experiment setup

Two basic components were made available for conducting this experiment: the Croatian Dependency Treebank in CoNLL (cf. Buchholz et al. 2006) format and the algorithm for extracting verb valency frame instances from it, i.e. the algorithm presented in (Agić et al. 2010).

The treebank, i.e. its 2.870 manually annotated sentences, is stored in the native TrEd feature structure (FS) format. Using TrEd, we converted the treebank into the Czech sentence tree structure (CSTS) format and then easily translated this format into the CoNLL format by simple regular expressions. Further, we implemented a script for CoNLL token validation and filtered out sentences with invalid tokens. The results of this filtering are given in Table 1: token encoding issues invalidated 171 sentences and thus left a total of 66.930 tokens available for the experiment. The aforementioned token encoding issues were mainly caused by missing escape sequences for decimal numbers within FS-formatted sentences and are currently being corrected. However, out of the 2.699 valid sentences available in CoNLL format, at the moment of conducting this experiment, only 1242 were already double-checked by expert linguists dealing with adapting the PDT formalism to the specifics of Croatian syntax. Therefore, once again as indicated by Table 1, only 1.242 sentences and 29.892 tokens were used here.

Feature	Treebank	This experiment
Sentences	2699	1242
Tokens	66930	29892
Lemmas	8995	5501
MSD tags	798	649
Analytical functions	80	65

Table 1. Treebank stats

The previously mentioned extraction algorithm – described in more detail by (Agić et al. 2010) – was also modified for purposes of this experiment. Its previous version was designed to detect only the verbs annotated with analytical functions *Pred*, *Pred_Co* and *Pred_Pa* and descend one level down the dependency tree to retrieve subjects (*Sub*), objects (*Obj*), adverbs (*Adv*) and nominal predicates (*Pnom*) or two levels down to retrieve the same tokens (annotated as *Sub*, *Obj*, *Adv*, *Pnom*) introduced by using subordinate conjunctions (*AuxC*) and prepositions (*AuxP*). Here, we adapted the algorithm to retrieve any verbs found in the dependency structure, regardless of their respective analytical functions and position within the dependency trees. The adaptation itself is implemented in order to raise the recall of the algorithm (while still maintaining its precision by not changing the simple set of descending rules), i.e. to retrieve as much verbs as possible given the limited size of the treebank sample used in the experiment.

```
              biti  [ dovršiti dovršena Vmps-sfp Pnom ]     [ studija studija Ncfsn Sb ]
       (biti Obj)  [ dovršiti dovršena Vmps-sfp Pnom PAT ] [ studija studija Ncfsn Sb ACT ]

 djelovati(djeluje  [ neozbiljno Neozbiljno Rnp Adv ]      [ odustajanje odustajanje Ncnsn Sb ]
            Pred)  [ neozbiljno Neozbiljno Rnp Adv MANN ] [ odustajanje odustajanje Ncnsn Sb ACT ]

       oslobodit i  [ nikada Nikada Rt Adv ]     [ zloduh zloduha Ncmsg Obj ]
 (oslobodili Pred)  [ nikada Nikada Rt Adv THL ] [ zloduh zloduha Ncmsg Obj PAT ]

          postati  [studij studiji Ncmpn Sb]     [fakultet fakultet Ncmsn Obj]
   (postali Pred)  [studij studiji Ncmpn Sb ACT] [fakultet fakultet Ncmsn Obj PAT]

        postojati  [ objektivno Objektivno Rnp Adv ]      [ problem problem Ncmsn Sb ]
 (postoji Pred_Co)  [ objektivno Objektivno Rnp Adv MANN ] [ problem problem Ncmsn Sb ACT ]

         prerasti  [ šuma u->šumu Spsa->Ncfsa AuxP->Adv ]
(prerastao ExD_Co)  [ šuma u->šumu Spsa->Ncfsa AuxP->Adv EFF ]

          započeti  [ proces Proces Ncmsn Sb ]     [ već već Rt Adv ]
 (započeo Pred_Co)  [ proces Proces Ncmsn Sb ACT ] [ već već Rt Adv MANN ]

       zaustaviti  [ oni ih Pp3-pa--y-n-- Obj ]     [ dolina u->dolini Spsl->Ncfsl AuxP->Adv ]
  (zaustavio Atr)  [ oni ih Pp3-pa--y-n-- Obj PAT ] [ dolina u->dolini Spsl->Ncfsl AuxP->Adv LOC ]
```

Figure 1. An example verb valency frame instance and its annotation

The algorithm was run on the treebank sample, extracting 2930 valency frame instances. Tectogrammatical functors were afterwards manually assigned to the extracted wordforms, as illustrated in Figure 1. A total of 936 frame instances were annotated for 424 different verbs. The following section presents the results obtained by counting co-occurrences of tectogrammatical functors and valency frames on the one side and verbs, morphosyntactic tags and analytical functions on the other.

In order to annotate verbal frames we used a set of functors used to describe verb valency, namely 5 argument functors and functors for 32 free modification functors. This is the list of free modification we used:

(1) Argument functors: *ACT* (actor), *PAT* (patient), *ADDR* (addressee), *ORIG* (origin), *EFF* (effect)

(2) Temporal functors: *TWHEN* (when), *TFHL* (for how long), *TFRWH* (from when), *THL* (how long), *THO* (how often), *TOWH* (to when), *TPAR* (temporal parallel), *TSIN* (since when), *TTILL* (till)

(3) Locative and directional functors: *DIR1* (where from), *DIR2* (which way), *DIR3* (where to), *LOC* (where)

(4) Functors for causal relations: *AIM* (purpose), *CAUS* (cause), *CNCS* (concession), *COND* (condition), *INTT* (intention)

(5) Functors for expressing manner: *ACMP* (accompaniment), *CPR* (comparison), *CRIT* (criterion), *DIFF* (difference), *EXT* (extent), *MANN* (manner), *MEANS* (means), *REG* (regard), *RESL* (result), *RESTR* (restriction)

(6) Functors for specific modifications: *BEN* (benefactor), *CONTRD* (contradiction), *HER* (heritage), *SUBS* (substitution)

This set of functors was chosen because we believe that they are sufficient to capture and represent main syntactic and semantic relations within sentences covering major morphosyntactic functions such as subject, object and various types of adverbials. On the other hand, a similar set of functors is used in lexica dealing exclusively with verb valency, such as CROVALLEX (Mikelić Preradović et al. 2009), developed for Croatian.

4. Results and discussion

Seven distributional properties were obtained by analyzing the previously presented manual annotation of valency frame instances within our testing framework:

(1) frequency of applied tectogrammatical functors,

(2) frequency of verb lemmas,

(3) frequency of functor n-grams, i.e. valency frames,

(4) distribution of valency frames from the previous distribution according to the verb they represent,

(5) distribution of morphosyntactic tags across functors,

(6) distribution of syntactic, i.e. analytical functions across functors and

(7) the previous two distributions combined, i.e. the distribution of pairs of analytical functions and morphosyntactic tags across tectogrammatical functors.

These results are presented in a somewhat compressed form in tables 2, 3 and 4 and brief interpretation of the presented data is given further in the text.

Table 2 provides the frequency of functors used in annotation and appears to be rather straightforward and expected. Namely, the most frequent functors are *PAT* (Patient), *ACT* (Actor) and *LOC* (Location), accounting for more than 70% of all the assigned functors[1]. The counts for the Actor functor should therefore be incremented by the number of occurrences of the Patient functor in Table 2. Additionaly, due to the FGD formalism, every argument following the Actor in two- or three-argument frames is implied to be labeled as Patient regardless of its cognitive content.

Functor	Count	Percent
PAT[1]	773	36.07
ACT	637	29.72
LOC	128	5.97
TWHEN	115	5.37
MANN	114	5.32
ADDR	43	2.01
CAUS	35	1.63
MEANS	26	1.21
DIR3	24	1.12
CRIT	23	1.07
AIM	22	1.03
THO	22	1.03
Other	181	8.45

Table 2. Functor frequency

[1] It should be noted that the overall number of wordforms annotated as Patient (PAT) should not in any case be larger than the number for Actor (ACT); the Actor is thus implied by the Patient within all the frames, even though it may not explicitly occur.

In Table 3, the actual frames – sequences of tectogrammatical functors occurring with a verb – are counted. In this presentation form, we do not display the frames as attached to specific verbs, as e.g. in Figure 3. Rather, we simply display the frequencies of the frame types independently. The table indicates that the Actor-Patient (*ACT PAT*) frame is the most frequent one, once again taking into account the emphasized note regarding the Patient functor and frame (*PAT*) from the previous table[1].

Table 4 represents a key point of our experiment. It is extracted from an obtained distribution of pairs of analytical functions and morphosyntactic tags across the tectogrammatical functors. Basically, for each functor, occurrences of specific ordered pairs (analytical function, morphosyntactic tag) were counted. These occurrence maps were assigned to the functors. The distribution, as illustrated by the table, can be used directly in writing down simple rules for the inference of tectogrammatical functors from wordforms in unseen (but morphosyntactically annotated and dependency-parsed) text. In the table, for purposes of illustration, the distributions are given just for the six most frequent tectogrammatical functors (Actor, Patient and Locative) and ten most frequent pairs of morphosyntactic tags and analytical functions.

Frame	Count	Percent
ACT PAT	250	26.71
PAT[1]	157	16.77
ACT PAT TWHEN	30	3.21
ACT MANN PAT	23	2.46
ACT ADDR PAT	20	2.14
ACT LOC	20	2.14
ACT LOC PAT	20	2.14
MANN PAT	17	1.82
ACT CAUS PAT	16	1.71
ACT MANN	13	1.39
LOC PAT	12	1.28
ADDR PAT	11	1.18
Other	347	37.07

Table 3. Frame frequency

ACT (Actor)			PAT (Patient)			LOC (Locative)		
A-fun	**MSD**	**%**	**A-fun**	**MSD**	**%**	**A-fun**	**MSD**	**%**
Sb	Ncmsn	14.91	Obj	Ncfsa	11.25	(AuxP) Adv	(Spsl) Ncfsl	21.88
Sb	Np-sn	13.50	Obj	Ncmsa	9.18	(AuxP) Adv	(Spsl) Ncmsl	16.41
Sb	Ncfsn	12.87	Pnom	Ncmsn	5.69	(AuxP) Adv	(Spsl) Npmsl	10.16
Sb	Ncmpn	9.89	Obj	Ncmpa	4.53	(AuxP) Adv	(Spsl) Ncnsl	8.59
Sb	Npfsn	5.65	Obj	Vmn*	4.40	(AuxP) Adv	(Spsl) Npfsl	8.59
Sb	Pi-mpn--n-a--	4.71	Obj	Ncnsa	3.75	(AuxP) Adv	(Spsl) Ncmpl	5.47
Sb	Ncfpn	3.30	Obj	Ncfpa	3.49	(AuxP) Adv	(Spsl) Ncfpl	3.91
Sb	Ncnsn	2.98	Pnom	Ncfsn	2.72	Adv	Rl	3.13
Sb	Pi-msn--n-a--	2.51	(AuxC) Obj	(Css) Vmip3s	2.07	Adv	Css	1.56
Sb	Pi-fsn--n-a--	1.88	Obj	Ncmsn	1.81	(AuxP) Adv	(Spsg)Ncmsg	1.56
TWHEN (Temporal when)			**MANN (Manner)**			**ADDR (Addressee)**		
A-fun	**MSD**	**%**	**A-fun**	**MSD**	**%**	**A-fun**	**MSD**	**%**
Adv	Rt	30.43	Adv	Rnp	40.35	Obj	Ncfsd	13.95
(AuxP) Adv	(Spsl) Ncmsl	12.17	Adv	Rn	20.18	Obj	Ncmpd	9.30
Adv	Ncfsg	5.22	Adv	Css	5.26	Obj	Pp3msd--y-n--	9.30
(AuxP) Adv	(Spsg) Ncmsg	5.22	Adv	Rt	3.51	Obj	Ncmsd	6.98
Adv	Ncmpg	4.35	(AuxP) Adv	(Spsi) Ncnsi	3.51	Obj	Ncnsd	6.98
Adv	Ncfsi	3.48	(AuxP) Adv	(Spsl) Ncfsl	3.51	(AuxP) Adv	(Spsi) Ncnsi	4.65
Adv	Ncmsi	3.48	Adv	Rk	1.75	Obj	Np-sd	4.65
(AuxP) Adv	(Spsl) Ncfsl	3.48	Adv	Rnc	1.75	(AuxP) Adv	(Spsa) Ncmsa--n	2.33
Adv	Ncmsg	2.61	(AuxP) Adv	(Spsi) Ncmsi	1.75	(AuxP) Adv	(Spsa) Ncmsg	2.33
(AuxP)Adv	(Spsl) Ncnsl	2.61	Adv	Afpmsn-	0.88	(AuxP) Adv	(Spsa) Px--sa--npn--	2.33

Table 4. Distribution of (analytical function, MSD) pairs for the most frequent functors

A simple example of utilizing the data in Table 4 for the inference of functors in unseen, but preprocessed text would be the one for assigning the Actor (*ACT*) functor. Namely, if a wordform (1) annotated as a noun in the nominative case (*N…n*) and (2) with an assigned syntactic function of subject (*Sb*) is encountered, the Actor functor should also be assigned to it. Such rules could also assign confidence measures to the outputted functors; these measures could be based e.g. on the occurrence percentages given in Table 4. Once again taking the Actor functor as an example, the confidence of assigning this functor to a {subject, nominative} noun would be at least 60 percent, a number derived by adding the percentages of {subject, nominative} entries in the table.

5. Conclusions and future work

In this experiment, we have designed and implemented one possible approach to semi-automatic extraction of a valency frame lexicon for Croatian verbs and also to the refinement of existing lexicons by using the Croatian Dependency Treebank as an underlying resource. We have automatically extracted 2930 verb valency frame instances and annotated 936 frames with tectogrammatical functors selected from the FGD formalism. We analyzed these annotations and provided two important results: (1) the distribution of valency frames for each of the encountered verbs and (2) the distribution of analytical functions and morphosyntactic tags for each of the tectogrammatical functors. The first result directly enables the enrichment of existing valency lexicons, such as CROVALLEX (Mikelić Preradović et al. 2009), while the second result enables the implementation of a rule-based system for automatic assignment of tectogrammatical functors to morphosyntactically tagged and dependency-parsed unseen text.

We divide our future research plans in the track of valency frame extraction into several directions. In the first one we will try (a) to implement and evaluate the previously mentioned rule-based system for assigning semantic roles to wordforms in unseen text and (b) to investigate the possibilities of semi-automatic enrichment of CROVALLEX with the verb valency frames extracted in this experiment. In the second one information on verb valency could also be utilized for the enrichment of Croatian WordNet (Raffaelli et al. 2008), namely by adding the valency frames to the verbs it encodes (cf. Pala & Sedlácek 2005)). Also, as implied in the previous sections, the treebank itself requires both enlargement and enhancements. Extensive efforts are currently underway with respect to these goals.

This procedure of automatic detection of valency frames will be used also in several other projects dealing with factored SMT (e.g. ACCURAT) where valency information will represent one of the layers of additional linguistic annotation that will be taken into account when developing translation models.

We also consider various approaches to dependency parsing of Croatian. Future research plans for this line of research are rather extensive. Regarding dependency parsing of Croatian by using the Croatian Dependency Treebank, we shall undergo various research directions in order to increase overall parsing accuracy.

In the first run we should investigate the performance of all freely available state-of-the-art data-driven dependency parsers. For example, in Table 5, the baseline scores obtained by using the linear-time algorithms of the MaltParser system (Nivre et al. 2007) are presented as an illustration of improvement possibilities with respect to the rather poor accuracy scores obtained in the trial run.

In the second run fine-tuning of all the available parameters for these should be investigated with respect to the specific properties of Croatian. Experiment with combining parsers and different parsing settings along the lines of experiments with the Index Thomisticus treebank (Passarotti & Dell'Orletta 2010) should also be conducted. Specifically, we would like to look into the possibilities of hybridization of the before-mentioned state-of-the-art data-driven parsers by linking them with language specific resources such as valency lexicons, following e.g. (Zeman 2002), being that a valency lexicon of Croatian verbs (CROVALLEX) already exists and the basic idea of verb valency (and valency in general) actually implies and constrains the dependency relations within a sentence. These research paths will be accompanied by a more elaborate investigation into all the different variables, i.e. treebank-encoded properties of Croatian language influencing the various aspects of dependency parsing accuracy.

Metric	Nivre eager	Nivre standard	Stack projective
Labeled attachment (LAS)	58.29±0.67	55.07±0.84	57.58±0.68
Unlabeled attachment (UAS)	67.91±0.59	67.31±0.77	67.49±0.64
Attachment of labels (LA)	70.85±0.45	64.73±0.69	72.36±0.54

Table 5. Baseline dependency parsing scores (MaltParser)

Acknowledgements

The research within the project ACCURAT leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013), grant agreement no 248347. This work was also supported by the Ministry of Science, Education and Sports, Republic of Croatia, under the grants No. 130-1300646-1002, 130-1300646-1776 and 130-1300646-0645.

References

Agić Ž, Tadić M, Dovedan Z. (2008). Improving Part-of-Speech Tagging Accuracy for Croatian by Morphological Analysis. Informatica, 32:4, pp. 445-451.

Agić Ž, Šojat K, Tadić M. (2010). An Experiment in Verb Valency Frame Extraction from Croatian Dependency Treebank. Proceedings of the 32nd International Conference on Information Technology Interfaces, Zagreb, SRCE University Computer Centre, University of Zagreb, 2010. pp. 55-60.

Buchholz S, Marsi E. (2006). CoNLL-X Shared Task on Multilingual Dependency Parsing. Proceedings of the 10th Conference on Computational Natural Language Learning (CoNLL), New York, NY, pp. 149-164.

Džeroski S, Erjavec T, Ledinek N, Pajas P, Žabokrtský Z, Žele A. (2006). Towards a Slovene Dependency Treebank. Proceedings of Fifth International Conference on Language Resources and Evaluation, LREC'06, 24-26 May 2006. Genoa.

Erjavec T. (2004). Multext-East Version 3: Multilingual Morphosyntactic Specifications, Lexicons and Corpora. Proceedings of the Fourth International Conference on Language Resources and Evaluation. ELRA, Lisbon-Paris 2004, pp. 1535-1538.

Hajič J, Böhmová A, Hajičová E, Vidová Hladká B. (2000). The Prague Dependency Treebank: A Three-Level Annotation Scenario. Treebanks: Building and Using Parsed Corpora, Amsterdam, Kluwer, 2000. See also URL http://ufal.mff.cuni.cz/pdt2.0/

Hajič J, Panevová J, Urešová Z, Bémová A, Pajas P. (2003). PDTVALLEX: Creating a Large-coverage Valency Lexicon for Treebank Annotation. Proceedings of the Second Workshop on Treebanks and Linguistic Theories, Vaxjo University Press, 2003, pp. 57-68.

Kübler S, McDonald R, Nivre J. (2009). Dependency Parsing. Synthesis Lectures on Human Language Technologies, Morgan\&Claypool Publishers, 2009.

Mikelić Preradović N, Boras D, Kišiček S. (2009). CROVALLEX: Croatian Verb Valence Lexicon. Proceedings of the 31st International Conference on Information Technology Interfaces, pp. 533-538. See URL http://cal.ffzg.hr/crovallex/index.html.

Nivre J, Hall J, Nilsson J, Chanev A, Eryigit G, Kübler S, Marinov S, Marsi E. (2007). MaltParser: A Language-Independent System for Data-Driven Dependency Parsing. Natural Language Engineering, 13(2), 95-135.

Nivre J, Hall J, Kübler S, McDonald R, Nilsson J, Riedel S, Yuret D. (2007). The CoNLL 2007 Shared Task on Dependency Parsing. Proceedings of the CoNLL Shared Task of EMNLP-CoNLL 2007, Prague, Czech Republic, pp. 915-932.

Pajas P. (2000). Tree Editor TrEd, Prague Dependency Treebank, Charles University, Prague. See URL http://ufal.mff.cuni.cz/~pajas/tred.

Pala K, Sedlácek R. (2005). Enriching WordNet with Derivational Subnets. Proceedings of CICLing 2005, pp. 305-311.

Passarotti M, Dell'Orletta F. (2010). Improvements in Parsing the Index Thomisticus Treebank. Revision, Combination and a Feature Model for Medieval Latin. Proceedings of the Seventh conference on International Language Resources and Evaluation, ELRA, 2010.

Raffaelli I, Tadić M, Bekavac B, Agić Ž. (2008). Building Croatian WordNet. Proceedings of the 4th Global WordNet Conference, Szeged, Global WordNet Association, 2008, pp. 349-359.

Sgall P, Hajičová E, Panevová J. (1986). The Meaning of the Sentence in Its Semantic and Pragmatic Aspects. Dordrecht, D. Reidel Publishing Company.

Šojat K, Vučković K, Tadić M. (2010). Extracting verb valency frames with Nooj. Finite State Language Engineering: NooJ 2009 International Conference and Workshop, Touzeur, Centre de Publication Universitaire, 2010. pp. 231-241.

Tadić M. (2002). Building the Croatian National Corpus. Proceedings of the 3rd International Conference on Language Resources and Evaluation, ELRA.

Tadić M. (2007). Building the Croatian Dependency Treebank: the initial stages. Suvremena lingvistika, 63, pp. 85-92.

Zeman D. (2002). Can Subcategorization Help a Statistical Dependency Parser? Proceedings of the 19th International Conference on Computational Linguistics.

PREPOSITIONAL PHRASES IN BULGARIAN[1]

Radka Vlahova, Atanas Atanasov

Department of Bulgarian Language, Faculty of Slavic Studies, Sofia University "St. Kliment Ohridski"
15 Tzar Osvoboditel, 1509 Sofia, Bulgaria
{rvlahova, atanasow}@gmail.com

ABSTRACT

In this paper we discuss Bulgarian prepositional phrases in VP which are not in their typical syntactic and semantic positions. The main goal is to investigate the types of predicates which allow these PPs, especially when they are in argument-adjunct or in predicative position. We have already analyzed the phrases with the preposition "с" (with) in FASSBL 2006, so here we are going to examine some phrases with other prepositions and will try to find out a consistency in their behavior.

1. Introduction

It is well known that the constructions with prepositional phrases in Bulgarian are discussed as a result of the loose of the morphological case declension. R. Nitsolova classifies them as arguments, argument-adjuncts and adjuncts, and the prepositions are presented as predicative (formal) and non-predicative, according to their semantics. The non-predicative (which semantics is defined by the predicate) connect arguments with the predicate and they are controlled by the predicate, since the predicative prepositions connect adjuncts with the predicate (they are called predicative as they involve an implicit predicate in the semantic structure of the sentence, which is not expressed in the surface syntactic level).

1. Positions of the prepositional phrase (PP)

The prepositional phrase in Bulgarian could in the following syntactic positions:

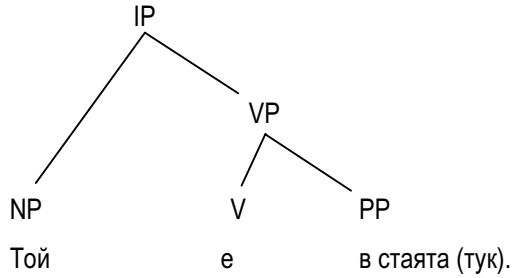
- part of a verb phrase (VP) – *говоря с нея*;
- part of a noun phrase (NP) – *чаша за кафе*;
- part of an adjective phrase (AP) – *способен на всичко*;
- part of an adverbial phrase (AdvP) – *независимо от това*.

Here we will focus on the firs case - when the PP is part of VP. There are four possibilities for the PPs in this position – it could be a predicative, an argument of the predicate, an argument-adjunct or an adjunct. The objects of our observation are the prepositional phrases in predicative and in argument-adjunct position.

2. Prepositional phrase in predicative position

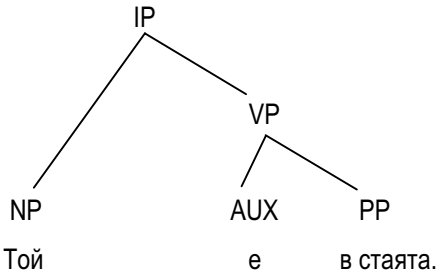
The question about the phrases of the type AUX + PP (auxiliary verb and prepositional phrase) is still not quite clarified in the Bulgarian linguistics. Some authors think that in such sentences, containing PP which could be replaced by an adverbial phrase, the predicator is just the verb *съм (to be)* – therefore *съм* in these constructions is not auxiliary verb, it is a synonymous of a verb, expressing existence:

[1] This paper is a part of the research project **Mathematical Logic and Computational Linguistics: Development and Permeation** (2009-2011). The financial support is granted under **Contract No. BG051PO001-3.3.04/27** of 28 August 2009 within the Operation ***Support to the development of PhD students, post-doctoral students, post-graduate students and young scientists*** of the General Directorate Structural Funds and International Educational Programmes with the Ministry of Education, Youth and Science.



In this case the prepositional phrase is an adjunct (Penchev 1998).

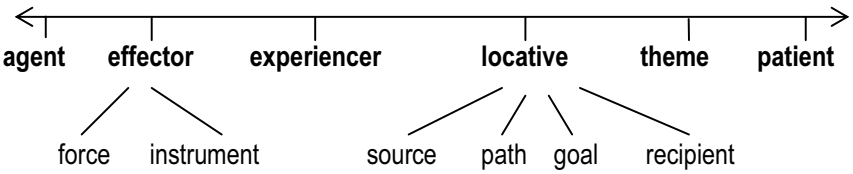
Other linguists (Brezinski 2001) claim that here the whole phrase *е в стаята* is the predicate, i.e. the PP is in predicative position:



We support the second opinion – the PP is a part of the predicate and the syntactic unit V –> AUX PP as a whole assigns a semantic role to its arguments.

3. Semantic roles

The next step is to find out what semantic roles assign the predicates containing auxiliary verb and prepositional phrases to their arguments. Here, as a theoretical framework, we accept Van Valin's approach, stated in the Role and Reference Grammar (Van Valin 1997), where the relations between the predicates and their arguments are presented with the following scheme:



Usually the predicates with PP have only one argument (in subject position). Our goal is to classify them according to their semantic class. They could mean genesis, age, time (date), location, value (quantification), language, possession, psychological condition etc. Nevertheless sometimes the semantic role of the argument is *experiencer* (*Не съм в настроение* – V: [$NP_{experiencer}$_]) or *recipient* (in the Role and Reference Grammar this role is equal to beneficent: *Той е с предимство* – V: [$NP_{recipient}$ _]), in most of the cases its role is *theme*:

Бъдещето е пред нас. (V: [NP_{theme}_])

Ние сме на власт. (V: [NP_{theme}_])

Кой е в колата? (V: [NP_{theme}_])

Тя е от свещи. (V: [NP_{theme}_])

Одобрението е за аматьори. (V: [NP_{theme}_])

Това е за теб. (V: [NP_{theme}_])

When the predicate requires more then one argument again, in general, the subject is *theme* in combination with *experiencer* or *locative*:

Тази музика не ми е на сърцето. (V: [NP_{theme}_$PP_{experiencer}$])

Къщата е на два километра от града. (V: [NP_{theme}_$PP_{locative}$])

In some rare cases the predicate could have an argument with the role *agens*. For example the phrase with the preposition *за (with)* in predicative position could mean goal (purpose) -- *за добро е, за спомен е, за смях стана, оказа се за пример*: *Тази книга не става за четене.; Филмът не е за изпускане.; Всичко ли е за продан.; Сюжетът е за израстването.* The last sentence can not have another argument, while the preceding three ones could be discussed as a result of transformations (from active to passive voice). As a result they admit a second argument, which semantic role is *agens* and the subject argument again is *theme*.

That's why our future efforts are concentrated on the more complex investigation of the semantic role theme.

Another case of predicative name are the sentences with the so called "small clause":

*Имаха го **за** един от най-добрите специалисти в Европа.*

*Нямам комплекси и не се мисля **за** по-низше създание от мъжа", разсъждава Нина.*

*Няколко месеца по-късно Стоев назначи Томчев **за** шеф на специално разкритото за него отделение за детска ортопедия и конфликтът между двамата изчезна.*

Here the small clause assigns a semantic role to its dropped subject, which is the object of the main predicate. This complicated semantic relation will also be investigated in our future work.

References

Bennett, D. C. 1975. *Spatial and Temporal Uses of English Prepositions: An Essay in Stratificational Semantics*. London: Longman.

Brezinski, S. 2001. *Vidove skazuemi.* In S. Koeva, ed., *Savremenni lingvistichni teorii. Pomagalo po sintaksis*. Plovdiv: Universitetsko izdatelstvo "Paisii Hilendarski".

Emonds, J. E. 1985. *A Unified Theory of Syntactic Categories*. Dordrecht: Foris.

Jackendoff, R. 1973. Base Rules for PPs. In S. R. Anderson and P. Kiparsky, eds., *A Festschrift for Morris Halle*, 345–356. New York: Holt, Rinehart, and Winston.

Jackendoff, R. 1983. *Semantics and Cognition*. Cambridge, Mass.:MIT Press.

Koopman, H. 2000. Prepositions, postpositions, circumpositions, and particles. In *The Syntax of Specifiers and Heads*, 204–260. London: Routledge.

Libert, A. R. 2006. *Ambipositions. LINCOM studies in language typology, 13*. LINCOM.

Nitsolova, R. 2008. Balgarska gramatika. Morfologia. Sofia: Universitetsko izdatelstvo "Sv. Kliment Ohridski".

Penchev, Y. 1998. *Syntax*. Plovdiv: Vechernik.

Rauh, G. 1991. *Approaches to Prepositions*. Tübingen: Gunter Narr.

Van Valin, R. D., Jr., & R. J. LaPolla. 1997. *Syntax: Structure, Meaning and Function*. Cambridge: Cambridge University Press.

SENTENCE CLASSIFICATION AND CLAUSE DETECTION FOR CROATIAN

Kristina Vučković*, Željko Agić*, Marko Tadić**

*Department of Information Sciences, **Department of Linguistics
Faculty of Humanities and Social Sciences, University of Zagreb
Ivana Lučića 3, 10000 Zagreb, Croatia
{kvuckovi, zeljko.agic, marko.tadic}@ffzg.hr

ABSTRACT

We present a method for classifying Croatian sentences by structure and detecting independent and dependent clauses within these sentences and provide its evaluation. A prototype system applying the method was implemented by using the NooJ linguistic development environment, both for purposes of this experiment and for further utilization in a prototype rule-based chunking and shallow parsing system for Croatian. With regards to pre-processing, we implemented and evaluated three different approaches to designing the system: (1) no pre-processing of input sentences, (2) automatic morphosyntactic tagging of sentences by using the CroTag stochastic tagger and (3) manual morphosyntactic annotation of input sentences. All three approaches were evaluated for sentence classification and clause detection accuracy in terms of precision and recall. The highest scoring system was the one using sentences with manually assigned morphosyntactic tags as input: an overall F1-measure of 0.861 (P: 0.928, R: 0.813). In the paper, a more detailed discussion of system design and experiment setup is provided, followed by a discussion of the obtained results and future research directions.

1. Introduction

Many natural language processing tools and complex natural language processing systems assembled by pipelining these tools demand certain methods of pre-processing the text input in order to operate at required levels of accuracy and efficiency or more generally, from a software engineering point of view, in order to meet the various functional and non-functional user requirements. One of the basic pre-processing tasks in language technologies is the segmentation of input text into paragraphs, sentences, tokens, etc. Here, we inspect the problem of sentence segmentation from a less common viewpoint. The problem of sentence segmentation – separating the input text into sentences – is well known to be resolved by using simple regular expressions with an accuracy of above 99 percent correctly detected sentence boundaries. However, building on top of e.g. (Boras 1998), we choose to inspect Croatian sentences from a more elaborate – both linguistically and computationally motivated – perspective. Namely, we do not seek solely to detect the sentence boundaries, i.e. to discover beginnings and endings of sentences, but also to (1) detect boundaries of clauses in complex sentences and (2) detect their type according to the grammatical classification of Croatian sentences. In this way, an implementation of such an analysis of input text written in Croatian may at the same time serve various linguistically motivated inquiries and also be used as a pre-processing module for more complex pipelined natural language processing systems, the latter being further elaborated in the following sections of the paper.

In Croatian, we classify sentences by their purpose or by their structure (Barić et al. 2005). By purpose, we differentiate between declarative, interrogative and exclamatory sentences. By structure, we classify the sentences into two major groups: (1) simple sentences and (2) complex sentences. Simple sentences contain only one predicate, paired with a subject only or with both a subject and object(s). Complex sentences are subcategorized into (1) independent complex, or compound sentences and (2) dependent complex sentences. There are six types of compound sentences where the independent clause is connected to the main clause by using a conjunction and being coordinated with the main clause. Based on the conjunction used, we can differentiate the following types of dependencies between the clauses: (1) constituent clauses, conjunctions {*i, pa, te, ni, niti*}; (2) disjunctive clauses, conjunction {*ili,ili…ili*}; (3) opposite clauses, conjunctions {*a, ali, nego, no, već*}; (4) exclusive clauses, conjunctions {*samo, samo što, jedino, jedino što, tek*}; (5) conclusive clauses, conjunctions {*dakle, zato, stoga*} and (6) explanatory clauses, conjunctions {*jer, ta, jerbo*}. Dependent complex sentences are made by connecting two or more clauses in such a manner in which the main clause is independent while all the other clauses depend on the main clause and cannot stand alone in a sentence. Namely, we distinguish predicate, subject, object, attribute, apposition and adverbial type of clauses.

This research is based on and developed keeping in mind the specific requirements of systems that have already been developed for parsing simple Croatian sentences consisting of a subject, verb, direct and indirect object, adverbial of time,

place and manner (Vučković et al. 2008, Vučković 2009). With this research we hope to extend the existing model to recognize the before-mentioned sentence structures, which includes both classifying sentences by structure and detecting clauses within them in order to parse Croatian sentences more efficiently.

In the following section, a system – implemented as a NooJ module – for sentence and clause detection and classification in Croatian texts is presented. Sections 3 and 4 present the experiment's plan and discuss the obtained results. We conclude by sketching possible future research directions, specifically in terms of pipelining the system presented here to the chunker and partial parser for Croatian (Vučković et al. 2008, Vučković et al. 2009, Vučković 2009, Vučković et al. 2010).

2. Detection and classification system

The sentence and clause detection and classification system for Croatian is implemented in the latest version of the NooJ linguistic development environment (Silberztein 2003, Silberztein 2008, Silberztein 2009, Silberztein 2010).

The main grammar for Croatian clause detection consists of two types of sub-graphs: main clauses (*Mainclause* and *Mainclause2*) and independent clauses (*IndependentClauses1*, *IndependentClauses2*) that may appear once before or any number of times after the main clause(s). The main grammar is displayed in Figure 1, as shown in NooJ.

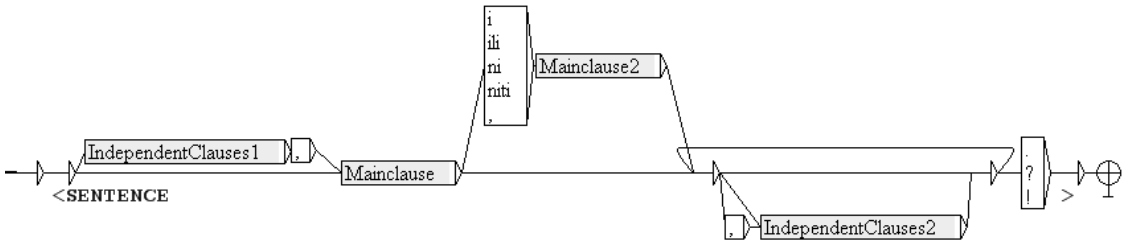


Figure 1. Top-level grammar of the system

As indicated by Figure 2, the main idea behind the grammar for clause detection is the presence of only one verb <V> or verb phrase <VP> and possibly any other phrase (noun phrase <NP> left part of Figure 3, prepositional phrase <PP>, adverb <R>, conjunction <C>, numeral <M>, pronoun <PRO>, preposition <S>) including the brackets and quotation marks as shown on the right part of Figure 3. This is true for all types of clauses (dependent and independent ones). The main difference between these two types of clauses is that the independent clauses do not depend on the verb from the main clause and may be recognized only according to the conjunction that they start with (see Figure 4). This is, however, not true of the dependant clauses with the exception of Adverbial dependant clause type. In order for the dependant clauses to be recognized, more than the conjunction recognition is necessary.

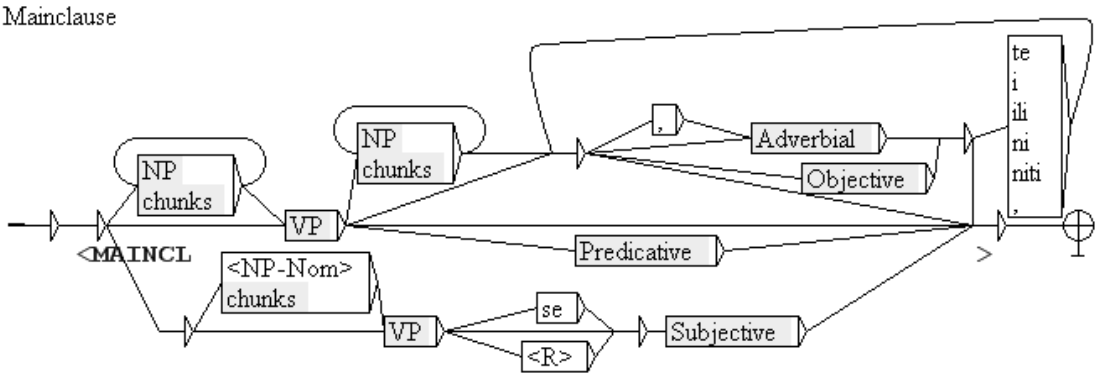


Figure 2. *Mainclause* grammar

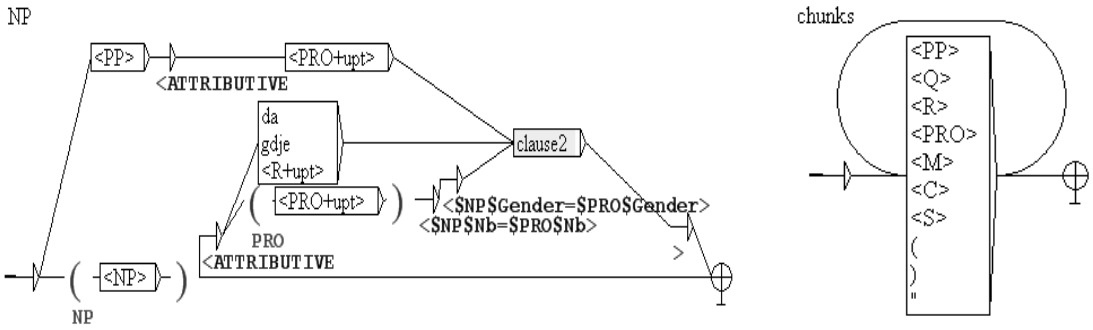


Figure 3. Grammars for detecting noun phrases and other chunks

Object dependant clause grammar has to check if the verb from the main clause takes as a complement an accusative noun phrase <NP+Acc>. Only then it continues to check if the type of the conjunction ({*da, gdje, kako, kuda, neka, kamo, koliko, kad*} or an interrogative pronoun) characteristic for this type of clauses is present before entering the clause itself.

Predicate dependant clause follows immediately after the verb from the main sentence and its grammar first checks if the verb from the main clause is any form of an auxiliary verb *to be* (hr. *biti*) or its negation *not to be* (hr. pres. 1p sg. *nisam*) since the entire predicate clause behaves as a nominal predicate to that verb. If this condition is met, the grammar checks if the conjunctions are of the predicate type ({*takav da, takva da, takvo da, tolik da, tolika da, toliko da*} or an interrogative pronoun) before entering the sub-graph for the clause recognition.

Subject dependant clause has a prerequisite of different form than the two previously explained clauses. It requires that any <NP> that may be present in the main sentence must not be in Nominative case <NP-Nom> since this entire clause behaves as a subject of the main clause. If this condition is met and if the conjunction of the clause is any from the following set ({*da, gdje, kako, kamo*} or an interrogative pronoun), the grammar will proceed to the clause itself.

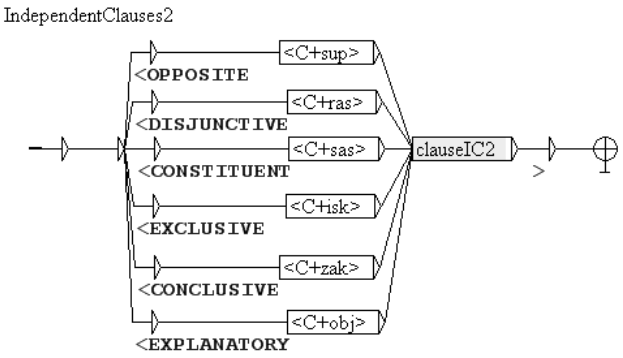


Figure 4. *IndependentClauses2* grammar

Adverbial dependant clause does not require any additional checks and it depends only on the conjunction that precedes it. The choice of a conjunction defines also the subtype of an adverbial clause in the following manner:

- Adverbial of time - {*kad, kada, dok, dokle, dočim, čim, jedva, tek, netom, pošto, kako, što, otkada, otkako, poslije nego, prije nego, kad god, dok god, dokle god, sve dok, samo dok, jedva što, tek što, istom što, netom što, nakon što, poslije nego što,pošto, prije nego što*}

- Adverbial of manner - {*kako, kao što, kao da, koliko, što, kano da, kanda*}
- Adverbial of place - {*gdje, kamo, kud, kuda, otkud, otkuda, odakle, dokle*}
- Adverbial of cause - {*jer, što, kad, kada, kako, budući da, jerbo, gdje, zašto, zato što, stoga što, zbog toga što, uslijed toga što, zahvaljujući tomu što*}
- Adverbial of condition - {(*i*) *ako, ukoliko, samo ako, samo da, samo kad*}

Attribute dependant clause may appear after any noun in the sentence. This is the reason why it is described together with the noun phrase <NP> (Vučković et al. 2008, Vučković 2009, Vučković et al. 2010). However, since the prepositional phrase <PP> ends with an <NP> chunk, it was necessary to add it into this sub-graph as well. Thus, if there is a <PP> followed by the interrogative pronoun, this pronoun may open the place for an entire clause. The similar goes for the noun phrase. If there is an <NP> followed by *da*, *gdje*, any interrogative type of adverb or any interrogative pronoun, this may open the place for an attribute clause. The difference between <NP> and <PP> being followed by a pronoun, is that in the first case number and gender agreement between the <NP> and a pronoun have to be checked (see Figure 3). However, this is, at the moment, not possible to check in the case of a <PP> chunk.

3. Experiment setup

At the time of conducting the experiment, no gold standard corpus containing the information on sentence types and clause boundaries was available. The Croatia Weekly 100 kw (CW100) corpus (cf. Tadić 2002, Vučković et al. 2008), being manually morphosyntactically annotated, lemmatized and XCES-encoded up to and including the sentence level, contained information on sentence boundaries. Thus, the CW100 corpus was used to manually assemble a small gold standard for the purposes of this specific experiment. More precisely, 200 sentences were chosen from the sentence pool of the CW100 corpus – 100 were assigned for the development stage, used for designing the system itself and other 100 were used for evaluation purposes. The sentences were chosen randomly, but the randomization process itself was biased towards selecting the sentences of average length, the average for CW100 being approximately 25 tokens, thus avoiding too short or too long sentences.

The detection and classification system was used in three different settings. In the first one, the 100 sentences of the testing set were provided as an input for the system without pre-processing. The second run used the CroTag stochastic morphosyntactic tagger (cf. Agić et al. 2008) to pre-process the sentences, with tagger accuracy of approximately 85 percent correctly annotated tokens, while the third run contained 100 percent accurate morphosyntactic annotation coupled with the sentences, as the annotation taken from the CW100 metadata for the sentences of the testing set. The outputs of these three systems were then evaluated and the results are discussed in the following section.

4. Results and discussion

Three sets of observations on the performance of the systems were made on their respective outputs. In this section, they are represented by tables 1, 2 and 3.

The first table gives an insight into the overall performance of the three systems in terms of their precision, recall and F1-measure as observed on the testing set. Recall was measured as a ratio of detected and existing clauses in the test sentences and it shows the system using manually tagged sentences (or an ideal tagger as a pre-processing module) as the top-performer with recall of 0.813, substantially higher than the other two systems. Interestingly enough, the table also shows that the noise introduced by the CroTag tagger and its decrease in tagging accuracy when compared with the ideal tagger actually decreases the detection performance even below the baseline set by the first system, i.e. the one using no pre-processing at all. However, when measuring system precision, utilizing the CroTag tagger does, in fact, somewhat improve the results and precision rises steadily from the first to the third system, even though this is not manifested in the respective F1-measures for the three systems, being that their differences for recall are more substantial than for precision.

	No tagging	CroTag	Manual tagging
Existing	289	289	289
Detected	211	190	235
Recall	0.730	0.657	**0.813**
Classified	187	169	218
Misclassified	23	19	17
Precision	0.890	0.899	**0.928**
F1-measure	0.802	0.759	**0.867**

Table 1. Precision and recall of the systems

Table 2 provides a type distribution for correctly detected classified sentence clauses across the systems. The table serves both as an indicator of the actual distribution of clauses in the testing set and as an addition to the general scores given in the previous table. Excluding the main clauses (MAIN), being that they expectedly dominate the distribution, object (OBJ), opposite (OPP), adverbial (ADV_CAUSE and others) and attribute (ATT) clauses are the most frequently encountered ones.

Sentence type	No tagging	CroTag	Manual tagging
ADV_CAUSE	14	12	16
ADV_COND	2	3	3
ADV_CONSEQ	0	0	1
ADV_MANNER	0	1	2
ADV_PLACE	0	0	1
ADV_TIME	2	1	2
ATT	10	8	15
CONST	10	9	9
DISJUNCT	1	1	1
EXPL	0	1	1
MAIN	85	79	95
OBJ	21	16	20
OPP	39	35	48
SUB	3	3	4

Table 2. Distribution of correctly classified sentences

Table 3 is basically a sentence classification confusion matrix, given summary for the three systems. The top row of the table is an indicator of the actual classification while the leftmost column lists all the misclassifications that occurred within the testing sample. As an illustration, the object clause (OBJ) was misclassified as main clause (MAIN) three times. Closely correlating with Table 2, attribute (ATT) and object (OBJ) clauses are most commonly misclassified, even though a larger testing set would surely provide a more informative confusion matrix.

	ADV_CAUSE	ADV_CONSEQ	ADV_MANNER	ADV_PURPOSE	ATT	CONSEQ	CONST	MAIN	OBJ	OPP	PURP	SUB
ADV	0	0	0	0	0	0	**2**	0	**2**	0	0	**2**
ADV_CAUSE	0	0	**1**	0	0	0	0	0	**1**	0	0	0
ADV_TIME	0	0	0	0	0	0	0	0	0	**1**	0	0
ATT	0	0	0	0	0	0	0	0	**3**	0	0	0
CONCL	0	**3**	0	0	0	0	0	**1**	0	0	0	0
CONST	0	**1**	0	0	0	**1**	0	0	0	0	0	0
EXPL	**2**	0	0	0	0	0	0	0	0	0	0	0
MAIN	**2**	0	0	0	**3**	0	**2**	0	**3**	**7**	0	0
OBJ	0	0	0	0	**2**	0	0	0	0	0	0	**1**
PRED	0	0	0	**3**	0	0	0	0	0	0	0	**1**
SUB	0	0	0	**3**	**4**	0	0	0	**5**	0	**3**	0

Table 3. Confusion matrix for sentence classification

Figure 5 shows disambiguation in the main clause and the independent opposite clause. They both have a dependant clause that can be interpreted fourfold as an object dependant clause or as an adverbial dependant clause of cause, manner or time. This ambiguity will be marked for all clauses that start with the conjunction *kako* if the verb from the main clause is a transitive verb that takes noun phrase in accusative as its complement. If the verb is intransitive, the clause will still be marked ambiguously as an adverbial clause of cause, manner or time only not as an object clause. There are more similar disambiguation problems due to the fact that different types of clauses share the same conjunctions.

Coordination of verbs is a problem that still needs to be revisited, especially in models where there are compound verb forms present in the sentence but the auxiliary verb "to be" is present for only one occurrence of the verb (usually the first one), while it is only implicitly transferred to the other verbs in a sentence. For example, in sentences such as: *Marko je pjevao u siječnju i glumio u travnju* (en. *Marko was singing in January and acting in April*). Nominal predicate also presents a problem at the present that will need to be looked more deeply into. For exmaple, such is a sentence: *Zakon bi nudio povlastice i ako je investicija usmjerena na ona područja Hrvatske koja su sada po gospodarskoj snazi ispod državnog prosjeka.* (en. *The law would offer benefits even if the investment is directed towards those areas of Croatia which are now concerning the economic power below the state percentage.*) with three clauses and two of which have a dislocated nominal predicates ('je …usmjerena', 'su… ispod').

```
<sentence>
  <maincl>Mesić je kazao
    <adverbial type="cause"><adverbial type="manner"><adverbial type="time">
          <objective>kako ne želi biti fikus</objective>
    </adverbial></adverbial></adverbial>
  </maincl>
  <opposite>dok je Račan ponovio
    <objective>
      <adverbial type="cause"><adverbial type="manner">
        <adverbial type="time">kako su se građani Hrvatske opredijelili za parlamentarnu demokraciju
      </adverbial></adverbial></adverbial>
    </objective>
  </opposite>.
</sentence>
```

```
<sentence>
  <maincl>Ivo Škrabalo je kasnije kazao
    <adverbial type="cause"><adverbial type="manner"><adverbial type="time">
          <objective>kako je Vijeće HRT-a jednoglasno dalo potporu programu direktora Galića</objective>
    </adverbial></adverbial></adverbial>
  </maincl>
  …
</sentence>
```

```
<sentence>
  …
  <opposite>a istaknuo je
    <subjective>
      <objective>
        <adverbial type="cause"><adverbial type="manner"><adverbial type="time">
          kako se radi i na donošenju Zakona o porezu na dohodak i Zakona o porezu na dobit
        </adverbial></adverbial></adverbial>
      </objective>
    </subjective>
  </opposite>.
</sentence>
```

Figure 5. Example sentences and annotation

Another problem is recognizing attribute dependant clause that starts with a prepositional phrase inside which there is an interrogative pronoun like '*za koje*' in the following example: *… jer je ponudio program promjena **za koje** su potrebni novi ljudi, …* (en. *… since he had offered a program of changes **for which** new people are needed, …*).

5. Conclusions and future work

We presented and evaluated a rule-based module developed in NooJ for sentence and clause classification and detection in Croatian texts. Assembling the module with the CroTag morphosyntactic tagger, we created three systems and evaluated them for precision and recall. The top performing system, i.e. the one using ideal morphosyntactic annotations for the input sentences, reached the F1-measure of 0.861 (precision: 0.928 recall: 0.813).

Further improvements of the system itself might include solving the problems with coordination of verbs, nominal predicates and attribute clases that start with a prepositional phrase (as explained in the previous section) but also detection of dependant clauses in other positions like when inserted between the main and its auxiliary verb or insertion of dependant clauses deeper into the sentence structure (beyond level 3 insertion that the system recognizes now) and combination of direct and indirect speech clauses. We are also planning an experiment with linking the system presented here with the rule-based chunker and shallow parser for Croatian (Vučković et al. 2008, Vučković et al. 2009, Vučković 2009, Vučković et al. 2010) in order to improve its overall accuracy on Croatian texts.

Acknowledgements

The research within the project ACCURAT leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013), grant agreement no 248347. This work was also supported by the Ministry of Science, Education and Sports, Republic of Croatia, under the grants No. 130-1300646-1776 and 130-1300646-0645.

References

Agić Željko, Tadić Marko, Dovedan Zdravko. (2008). Improving Part-of-Speech Tagging Accuracy for Croatian by Morphological Analysis. Informatica 32:4, pp. 445-451.

Barić Eugenija, Lončarić Mijo, Malić Dragica, Pavešić Slavko, Peti Mirko, Zečević Vesna, Znika Marija. (2005). Hrvatska gramatika, Školska knjiga, Zagreb.

Boras Damir. (1998). Teorija i pravila segmentacije teksta na hrvatskom jeziku. PhD Thesis, Department of Information Sciences, Faculty of Humanities and Social Sciences, University of Zagreb, Zagreb, 1998.

Silberztein Max. (2003). NooJ Manual. http://www.nooj4nlp.net/NooJ Manual.pdf, 2003.

Silberztein Max. (2008). Complex Annotations with NooJ. Proceedings of the 2007 International NooJ Conference. Cambridge Scholars Publishing, Newcastle, pp. 214-227.

Silberztein Max. (2009). Syntactic parsing with NooJ. Finite State Language Engineering: NooJ 2009 International Conference and Workshop, Centre de Publication Universitaire, pp. 177-189.

Silberztein Max. (2010). Disambiguation Tools for NooJ. Proceedings of the 2008 International NooJ Conference. Cambridge Scholars Publishing, Newcastle (in press).

Tadić Marko. (2002). Building the Croatian National Corpus. Proceedings of the 3rd International Conference on Language Resources and Evaluation, ELRA.

Vučković Kristina, Tadić Marko, Dovedan Zdravko. (2008). Rule Based Chunker for Croatian. Proceedings of the Sixth International Conference on Language Resources and Evaluation, Paris-Marrakech, 2008.

Vučković Kristina. Model parsera za hrvatski jezik. (2009). PhD Thesis, Department of Information Sciences, Faculty of Humanities and Social Sciences, University of Zagreb, Zagreb, 2009.

Vučković Kristina, Bekavac Božo, Dovedan Zdravko. (2009). SynCro – Parsing simple Croatian sentences. Finite State Language Engineering: NooJ 2009 International Conference and Workshop, Centre de Publication Universitaire, pp. 207-217.

Vučković Kristina, Agić Željko, Tadić Marko. (2010). Improving Chunking Accuracy on Croatian Texts by Morphosyntactic Tagging. Proceedings of the Seventh International Conference on Language Resources and Evaluation, European Language Resources Association, Valletta, pp. 1944-1949.

Vučković Kristina, Tadić Marko, Bekavac Božo. (2010). Croatian Language Resources for NooJ. Proceedings of the 32nd International Conference on Information Technology Interfaces, SRCE University Computer Centre, University of Zagreb, Zagreb, pp. 121-126.

Publisher Croatian Language Technologies Society / Faculty of Humanities and Social Sciences, University of Zagreb
• **For the publisher** Marko Tadić • **Print** Hitra produkcija knjiga, Zagreb

Supported by

Ministry of Science, Education and Sports
of the Republic of Croatia

Bulgarian Academy of Sciences

Croatian Language Technologies Society

ISBN 978-953-55375-2-6